```
   CCCCCCCCCCC  LLL              IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCC  LLL              IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCC  LLL              IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
CCC             LLL                 III      UUU         UUU        TTT        LLL
   CCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
```

FILEID**SETFILE

```
  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  FFFFFFFFFF    IIIIII    LL          EEEEEEEEEE
  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  FFFFFFFFFF    IIIIII    LL          EEEEEEEEEE
SS          EE              TT      FF              II      LL          EE
SS          EE              TT      FF              II      LL          EE
SS          EE              TT      FF              II      LL          EE
SS          EE              TT      FF              II      LL          EE
  SSSSSS    EEEEEEEE        TT      FFFFFFFF        II      LL          EEEEEEEE
  SSSSSS    EEEEEEEE        TT      FFFFFFFF        II      LL          EEEEEEEE
        SS  EE              TT      FF              II      LL          EE
        SS  EE              TT      FF              II      LL          EE
        SS  EE              TT      FF              II      LL          EE          ....
        SS  EE              TT      FF              II      LL          EE          ....
SSSSSSSS    EEEEEEEEEE      TT      FF          IIIIII      LLLLLLLLLL  EEEEEEEEEE  ....
SSSSSSSS    EEEEEEEEEE      TT      FF          IIIIII      LLLLLLLLLL  EEEEEEEEEE  ....


LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

```
;     1          0001  0 MODULE setfile  (
;     2          0002  0                 IDENT = 'V04-000',
;     3          0003  0                 ADDRESSING_MODE(EXTERNAL=GENERAL,
;     4          0004  0                                 NONEXTERNAL=LONG_RELATIVE)
;     5          0005  0                 ) =
;     6          0006  1 BEGIN
;     7          0007  1
;     8          0008  1 !
;     9          0009  1 !*****************************************************************************
;    10          0010  1 !*                                                                           *
;    11          0011  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    12          0012  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    13          0013  1 !*  ALL RIGHTS RESERVED.                                                      *
;    14          0014  1 !*                                                                           *
;    15          0015  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    16          0016  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    17          0017  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    18          0018  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    19          0019  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    20          0020  1 !*  TRANSFERRED.                                                              *
;    21          0021  1 !*                                                                           *
;    22          0022  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    23          0023  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    24          0024  1 !*  CORPORATION.                                                              *
;    25          0025  1 !*                                                                           *
;    26          0026  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    27          0027  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                    *
;    28          0028  1 !*                                                                           *
;    29          0029  1 !*                                                                           *
;    30          0030  1 !*****************************************************************************
;    31          0031  1 !
;    32          0032  1 !
;    33          0033  1 !++
;    34          0034  1 ! FACILITY:     Set File Command
;    35          0035  1 !
;    36          0036  1 ! ABSTRACT:
;    37          0037  1 !
;    38          0038  1 !       This module processes the Set File command.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:
;    41          0041  1 !
;    42          0042  1 !       Vax native, privileged user mode
;    43          0043  1 !
;    44          0044  1 !--
;    45          0045  1 !
;    46          0046  1 ! AUTHOR:       Gerry Smith             CREATION DATE: 04-Aug-1981
;    47          0047  1 !
;    48          0048  1 ! MODIFIED BY:
;    49          0049  1 !
;    50          0050  1 !       V03-023 AEW0005         Anne E. Warner          24-Jul-1984
;    51          0051  1 !               Make /EXPIRATION_DATE and /GLOBAL_BUFFERS check if
;    52          0052  1 !               the qualifier is present before trying to get any values
;    53          0053  1 !               associated with it.  This is needed because these qualifiers
;    54          0054  1 !               are now negatable.
;    55          0055  1 !
;    56          0056  1 !       V03-022 BLS0303         Benn Schreiber          12-APR-1984
;    57          0057  1 !               Parse null string after parse in /enter code.
```

```
;  58          0058  1 ;
;  59          0059  1 ;       V03-021 AEW0004         Anne E. Warner          10-Apr-1984
;  60          0060  1 ;               Fix SET FILE/PROTECTION so that it handles wildcarding.
;  61          0061  1 ;
;  62          0062  1 ;       V03-020 MCN0156         Maria del C. Nasr       08-Mar-1984
;  63          0063  1 ;               If the user specifies /VERSION=0, then it should default
;  64          0064  1 ;               to the maximum value: 32767.  Also, the maximum value is
;  65          0065  1 ;               32767, and not 65535.
;  66          0066  1 ;
;  67          0067  1 ;       V03-019 AEW0003         Anne E. Warner          28-Feb-1984
;  68          0068  1 ;               Add support for search lists.
;  69          0069  1 ;               -  remove related name block from RMS definitions.
;  70          0070  1 ;               -  add argument to LIB$FILE_SCAN
;  71          0071  1 ;
;  72          0072  1 ;       V03-018 LMP0191         L. Mark Pilant,         10-Feb-1984  16:27
;  73          0073  1 ;               Validate the value of the /OWNER qualifier.
;  74          0074  1 ;
;  75          0075  1 ;       V03-017 DAS0002         David Solomon           6-Feb-1984
;  76          0076  1 ;               Specify ACE$M_NOPROPAGATE for RMSJNLID ACE. Disable /JOURNAL.
;  77          0077  1 ;
;  78          0078  1 ;       V03-016 AEW0002         Anne Warner             15-Dec-1983
;  79          0079  1 ;               Add /PROTECTION qualifier with keywords /CONFIRM and
;  80          0080  1 ;               /LOG.
;  81          0081  1 ;
;  82          0082  1 ;       V03-015 JWT0139         Jim Teague              09-Nov-1983
;  83          0083  1 ;               Change the name of two of the RU fields; ensure that
;  84          0084  1 ;               we don't leave the file set with conflicting RU
;  85          0085  1 ;               attributes.
;  86          0086  1 ;
;  87          0087  1 ;       V03-014 AEW0001         Anne Warner             08-Nov-1983
;  88          0088  1 ;               Add /UNLOCK qualifier with keywords /CONFIRM and
;  89          0089  1 ;               /LOG.
;  90          0090  1 ;
;  91          0091  1 ;       V03-013 DAS0001         David Solomon           29-Jul-1983
;  92          0092  1 ;               Fold /AI_JOURNAL, /BI_JOURNAL, and /AT_JOURNAL into keywords
;  93          0093  1 ;               on the /JOURNAL qualifier. /JOURNAL keyword RUM is now
;  94          0094  1 ;               ONLY_RU. Add NEVER_RU keyword; a few journaling-related fixes.
;  95          0095  1 ;
;  96          0096  1 ;       V03-012 GAS0147         Gerry Smith             27-Jun-1983
;  97          0097  1 ;               Change the file attribute modification so that it is
;  98          0098  1 ;               done thru a IO$_MODIFY instead of simply during the
;  99          0099  1 ;               IO$_DEACCESS.  This is necessary for the case of
; 100          0100  1 ;               the version limit, since it cannot be changed on file
; 101          0101  1 ;               deaccess.
; 102          0102  1 ;
; 103          0103  1 ;       V03-011 GAS0141         Gerry Smith             17-Jun-1983
; 104          0104  1 ;               Signal all common qualifiers more completely, so that
; 105          0105  1 ;               the specified file can be found.
; 106          0106  1 ;
; 107          0107  1 ;       V03-010 KPL0002         Peter Lieberwirth       30-May-1983
; 108          0108  1 ;               Change JSB$S_JNLNAM to CJF$C_MXJNLNAML.
; 109          0109  1 ;
; 110          0110  1 ;       V03-009 KPL0001         Peter Lieberwirth       20-Apr-1983
; 111          0111  1 ;               Set journal names via new qualifiers AI_JOURNAL, BI_JOURNAL,
; 112          0112  1 ;               and AT_JOURNAL.  When marking the file for journaling,
; 113          0113  1 ;               write an RMSJNLID ACE.
; 114          0114  1 ;
```

```
;  115     0115  1 ;        V03-008 GAS0118         Gerry Smith             12-Apr-1983
;  116     0116  1 ;                Add the common qualifiers.
;  117     0117  1 ;
;  118     0118  1 ;        V03-007 GAS0112         Gerry Smith             30-Mar-1983
;  119     0119  1 ;                Convert to the new CLI interface, as well as a new
;  120     0120  1 ;                command dispatcher.
;  121     0121  1 ;
;  122     0122  1 ;        V03-006 TMK0001         Todd M. Katz            28-Feb-1983
;  123     0123  1 ;                If someone requested AI journalling on a file to be turned
;  124     0124  1 ;                off (/JOURNAL=NOAI) then turn it off. Currently, AI Journalling
;  125     0125  1 ;                will always be enabled whenever it is explicitely refered to
;  126     0126  1 ;                (/JOURNAL=AI or /JOURNAL=NOAI), and there is no way to disable
;  127     0127  1 ;                it.
;  128     0128  1 ;
;  129     0129  1 ;        V03-005 GAS0091         Gerry Smith             19-Oct-1982
;  130     0130  1 ;                Change input request for new CLD syntax.
;  131     0131  1 ;
;  132     0132  1 ;        V03-004 GAS0083         Gerry Smith             15-Jul-1982
;  133     0133  1 ;                Modify logic for RU journal option, to agree with
;  134     0134  1 ;                new definition.  The RUJNL bit used to have the opposite
;  135     0135  1 ;                sense of all other journal bits.  It now has the same sense.
;  136     0136  1 ;
;  137     0137  1 ;        V03-003 GAS0071         Gerry Smith              8-Apr-1982
;  138     0138  1 ;                If the writer count for a file is non-zero, don't allow
;  139     0139  1 ;                modification.  If /END is attempted on INDEXF.SYS, don't
;  140     0140  1 ;                allow it.
;  141     0141  1 ;
;  142     0142  1 ;        V03-002 GAS0068         Gerry Smith             31-Mar-1982
;  143     0143  1 ;                If a truncate is attempted on an indexed file, signal
;  144     0144  1 ;                an error.
;  145     0145  1 ;
;  146     0146  1 ;        V03-001 GAS0064         Gerry Smith             19-Mar-1982
;  147     0147  1 ;                Change check of qualifiers to include /GLOBAL_BUFFERS.
;  148     0148  1 ;
;  149     0149  1 ;        V03-005 GAS0050         Gerry Smith             22-Feb-1982
;  150     0150  1 ;                Only access the file header for something besides
;  151     0151  1 ;                /ENTER or /REMOVE.  Make the error messages for /ENTER
;  152     0152  1 ;                and /REMOVE more meaningful.  Change the /ENTER check for
;  153     0153  1 ;                same devices to use the DVI fields of the NAM blocks.
;  154     0154  1 ;
;  155     0155  1 ;        V03-004 GAS0047         Gerry Smith             15-Feb-1982
;  156     0156  1 ;                For SET FILE/ENTER, parse the new file name here, after
;  157     0157  1 ;                the old file name is available, so that stickiness can
;  158     0158  1 ;                be applied.
;  159     0159  1 ;
;  160     0160  1 ;        V03-003 GAS0038         Gerry Smith              2-Feb-1982
;  161     0161  1 ;                Add /GLOBAL_BUFFERS, the global buffer count for a
;  162     0162  1 ;                file.  Also, if the file is ODS1, then move the record
;  163     0163  1 ;                attributes to the location occupied in an ODS2 file.
;  164     0164  1 ;                This allows the BIND in routine SET_ATTRIBUTES to apply
;  165     0165  1 ;                to both kinds of file headers.
;  166     0166  1 ;
;  167     0167  1 ;        V03-002 GAS0026         Gerry Smith             18-Dec-1981
;  168     0168  1 ;                Use shared message file, and lower fatal messages to
;  169     0169  1 ;                simple error messages.
;  170     0170  1 ;
;  171     0171  1 ;        V03-001 GAS0024         Gerry Smith             14-Dec-1981
```

```
;  172     0172  1 !               Fix /LOG logic for /ENTER and /REMOVE
;  173     0173  1 !
;  174     0174  1 !       V03-001 MSH0001         Maryann Hinden          02-Dec-1981
;  175     0175  1 !               Change references to FIB$C_SIZE to FIB$C_LENGTH.
;  176     0176  1 !
;  177     0177  1 !       V03-001 GAS0021         Gerry Smith             30-Nov-1981
;  178     0178  1 !               Fix /VERSION, making FIB larger
;  179     0179  1 !
;  180     0180  1 !       V03-001 GAS0018         Gerry Smith             16-Nov-1981
;  181     0181  1 !               Split SET FILE into separate modules
;  182     0182  1 !
;  183     0183  1 !       V03-001 GAS0011         Gerry Smith             22-Sep-1981
;  184     0184  1 !               Fix wildcarding for /ENTER.  Add /END_OF_FILE
;  185     0185  1 !
;  186     0186  1 !       V03-002 GAS0012         Gerry Smith             30-Sep-1981
;  187     0187  1 !               Add /LOG and /CONFIRM
;  188     0188  1 !
;  189     0189  1 !**
```

```
;  191          0190  1 LIBRARY 'SYS$LIBRARY:LIB';
;  192          0191  1 LIBRARY 'SYS$LIBRARY:CLIMAC.L32';       ! CLI macros
;  193          0192  1
```

```
;  195     0193  1 FORWARD ROUTINE
;  196     0194  1     set$file : NOVALUE,                 ! Main routine for file
;  197     0195  1     get_quals,                          ! Get qualifiers
;  198     0196  1     set_attributes,                     ! Routine to set file attributes
;  199     0197  1                                         ! Common routines:
;  200     0198  1     unlock_action,                      ! Called to control each UNLOCK action
;  201     0199  1     check_privilege : NOVALUE,          ! Routine to check for privilege
;  202     0200  1     search_error,                       ! Where to go if file search fails
;  203     0201  1     file_error,                         ! Where to go if file error occur,
;  204     0202  1     setpro_action,                      ! Called to control each file PROTECTION
;  205     0203  1     prot_log_results,                   ! Called when user requests a log for PROTECTION
;  206     0204  1     expand_prot,                        ! Converts binary protection to ascii
;  207     0205  1     parse_class;                        ! Parses the protection of one user class
;  208     0206  1
;  209     0207  1 EXTERNAL ROUTINE
;  210     0208  1     parse_uic,                          ! Convert a UIC
;  211     0209  1     cli$present,                        ! Get qualifiers
;  212     0210  1     cli$get_value,                      ! Get values of qualifiers
;  213     0211  1     lib$cvt_dtb,                        ! Convert ASCII to numerical
;  214     0212  1     lib$cvt_time,                       ! Convert time to internal
;  215     0213  1     lib$file_scan,                      ! Routine to find next file
;  216     0214  1     lib$qual_file_parse,                ! Parse common qualifiers
;  217     0215  1     lib$qual_file_match,                ! Check for common qualifiers
;  218     0216  1     lib$confirm_act,                    ! Confirm action with user
;  219     0217  1     lib$get_command,                    ! Talk to SYS$COMMAND
;  220     0218  1     lib$unlock_file,                    ! Unlocks files
;  221     0219  1     sys$fao,                            ! Expands formatted messages
;  222     0220  1     sys$setprv,                         ! Set privileges for protection
;  223     0221  1     lib$set_file_prot;                  ! Set file protection
;  224     0222  1
;  225     0223  1 !
;  226     0224  1 ! Literal data definitions
;  227     0225  1 !
;  228     0226  1 LITERAL
;  229     0227  1     true = 1,
;  230     0228  1     false = 0;
;  231     0229  1
;  232     0230  1 MACRO
;  233     0231  1 !
;  234     0232  1 ! Macro definitions for fields in access control entries needed by RMS
;  235     0233  1 ! Journaling
;  236     0234  1 !
;  237     0235  1     id_ace$s_size = 32 %,               ! size in bytes of rmsjnlid ACE
;  238     0236  1     id_ace$t_label = 4,0,0,0 %,         ! volume label in rmsjnlid ACE
;  239     0237  1     id_ace$w_num = 16,0,16,0 %,         ! fid num in rmsjnlid ACE
;  240     0238  1     id_ace$w_seq = 18,0,16,0 %,         ! fid seq in rmsjnlid ACE
;  241     0239  1     id_ace$w_rvn = 20,0,16,0 %,         ! fid rvn in rmsjnlid ACE
;  242     0240  1     id_ace$q_time = 24,0,32,0 %,        ! time in rmsjnlid ACE
;  243     0241  1     ace$t_jnlnam = 4,0,0,0 %,           ! journal name string in jnl ACE
;  244     0242  1 !
;  245     0243  1 ! A) Macro to describe a string
;  246     0244  1 ! B) Macro to generate a quadword string descriptor
;  247     0245  1 ! C) Macro to generate the address of a string descriptor
;  248     0246  1
;  249     0247  1     PRIMDESC (str) = %CHARCOUNT (str), UPLIT (%ASCII str)%,
;  250     0248  1     INITDESC (str) = $BBLOCK [DSC$C_S_BLN] INITIAL (PRIMDESC (str))%,
;  251     0249  1     ADDRDESC (str) = UPLIT (PRIMDESC (str))%;
```

```
;  252          0250  1
;  253          0251  1
```

```
;  255          0252  1 !
;  256          0253  1 ! Define the data that is used by SET FILE
;  257          0254  1 !
;  258          0255  1 GLOBAL
;  259          0256  1     setfile$flags : BITVECTOR[32] INITIAL(0),    ! Qualifier bits word
;  260          0257  1     setfile$dflags : BITVECTOR[32] INITIAL(0),   ! DATA_CHECK options word
;  261          0258  1     setfile$jflags : BITVECTOR[32] INITIAL(0),   ! JOURNAL options word
;  262          0259  1     setfile$mflags : BITVECTOR[32] INITIAL(0),   ! Miscellaneous flags word
;  263          0260  1     setpro_prot    : WORD          INITIAL(0),   ! Contains /PROTECTION value
;  264          0261  1     setpro_mask    : WORD          INITIAL(0),   ! Contains /PROTECTION mask
;  265          0262  1     global_prot    : WORD,                       ! Command wide protection
;  266          0263  1     global_mask    : WORD,                       ! Command wide mask for protection
;  267          0264  1     exp_value      : $BBLOCK[8],                 ! Expiration date
;  268          0265  1     exte_value,                                  ! Extension quantity
;  269          0266  1     gbuf_value,                                  ! Global buffer value
;  270          0267  1     uic_value,                                   ! Owner uic
;  271          0268  1     group,                                       ! Group number
;  272          0269  1     member,                                      ! Member number
;  273          0270  1     vrsn_value,                                  ! Version limit
;  274          0271  1     rename_buf : VECTOR[nam$c_maxrss,BYTE],      ! Name buffer for /ENTER
;  275          0272  1     file_name : VECTOR[2],                       ! ENTER/REMOVE descriptor
;  276          0273  1     ai_jnl_name : VECTOR[cjf$c_mxjnlnaml,BYTE],  ! AI journal name
;  277          0274  1     at_jnl_name : VECTOR[cjf$c_mxjnlnaml,BYTE],  ! AT journal name
;  278          0275  1     bi_jnl_name : VECTOR[cjf$c_mxjnlnaml,BYTE],  ! BI journal name
;  279          0276  1     ai_jnl_desc : $BBLOCK[dsc$c_s_bln]           ! AI_JOURNAL descriptor
;  280          0277  1         PRESET( [dsc$a_pointer] = ai_jnl_name ),
;  281          0278  1     at_jnl_desc : $BBLOCK[dsc$c_s_bln]           ! AT_JOURNAL descriptor
;  282          0279  1         PRESET( [dsc$a_pointer] = at_jnl_name ),
;  283          0280  1     bi_jnl_desc : $BBLOCK[dsc$c_s_bln]           ! BI_JOURNAL descriptor
;  284          0281  1         PRESET( [dsc$a_pointer] = bi_jnl_name ),
;  285          0282  1     worst_error : $BBLOCK[4] INITIAL(ss$_normal),! Worst error reported
;  286          0283  1     conf_desc : $BBLOCK[dsc$c_s_bln],            ! Descriptor for /LOG/CONFIRM
;  287          0284  1
;  288          0285  1     oldpriv             : $BBLOCK[8],   ! Permanent priv's stored here
;  289          0286  1     newpriv             : $BBLOCK[8]    ! Mask describing system priv
;  290          0287  1                   PRESET ([prv$v_sysprv]=true), ! Initialize this bit
;  291          0288  1 !
;  292          0289  1 ! RMS storage
;  293          0290  1 !
;  294          0291  1     file_result : VECTOR[nam$c_maxrss,BYTE],     ! Resultant name string
;  295          0292  1     file_expanded : VECTOR[nam$c_maxrss,BYTE],   ! Expanded name string
;  296        P 0293  1     file_nam : $NAM(                             ! File name block
;  297        P 0294  1         ESA = file_expanded,
;  298        P 0295  1         ESS = nam$c_maxrss,
;  299        P 0296  1         RSA = file_result,                       ! File name after open
;  300          0297  1         RSS = nam$c_maxrss),
;  301        P 0298  1     file_fab : $FAB(                             ! FAB for file
;  302          0299  1         NAM = file_nam);                         ! Specify name block
;  303          0300  1
;  304          0301  1 !
;  305          0302  1 ! Declare the context block used by the common qualifiers
;  306          0303  1 !
;  307          0304  1 OWN
;  308          0305  1     context;
```

```
;  310          0306  1 !
;  311          0307  1 ! Declare the qualifier flag bits used by SET FILE
;  312          0308  1 !
;  313          0309  1 LITERAL
;  314        P 0310  1     $EQULST(QUAL_,,1,1,
;  315        P 0311  1             (backup,),
;  316        P 0312  1             (nobackup,),
;  317        P 0313  1             (confirm,),
;  318        P 0314  1             (data,),
;  319        P 0315  1             (eof,),
;  320        P 0316  1             (erase,),
;  321        P 0317  1             (noerase,),
;  322        P 0318  1             (expi,),
;  323        P 0319  1             (exte,),
;  324        P 0320  1             (gbuf,)
;  325        P 0321  1             (journal,),
;  326        P 0322  1             (log,)
;  327        P 0323  1             (nodi,),
;  328        P 0324  1             (owner,),
;  329        P 0325  1             (parent,),
;  330        P 0326  1             (protection,),
;  331        P 0327  1             (trunc,)
;  332        P 0328  1             (unlock,),
;  333        P 0329  1             (vrsn,)
;  334        P 0330  1             (enter,),
;  335        P 0331  1             (remove,),
;  336        P 0332  1             (quit,),
;  337        P 0333  1             (quit_mod,),
;  338        P 0334  1             (quit_rem,),
;  339        P 0335  1             (quit_ent,),
;  340        P 0336  1             (quit_protect),
;  341          0337  1             (quit_unlock,));
;  342          0338  1
;  343          0339  1 !
;  344          0340  1 ! Declare the DATA_CHECK option bits
;  345          0341  1 !
;  346          0342  1 LITERAL
;  347        P 0343  1     $EQULST
;  348        P 0344  1             (DATA_,,1,1,
;  349        P 0345  1             (read,),                    ! DATA_CHECK = READ
;  350        P 0346  1             (write,),                   ! DATA_CHECK = WRITE
;  351        P 0347  1             (noread,),                  ! DATA_CHECK = NOREAD
;  352          0348  1             (nowrite,));                ! DATA_CHECK = NOWRITE
;  353          0349  1
;  354          0350  1 !
;  355          0351  1 ! Declare the JOURNAL option bits
;  356          0352  1 !
;  357          0353  1 LITERAL
;  358        P 0354  1     $EQULST
;  359        P 0355  1             (JRNL_,,1,1,
;  360        P 0356  1             (ai,)
;  361        P 0357  1             (specified_ai,),
;  362        P 0358  1             (at,)
;  363        P 0359  1             (specified_at,),
;  364        P 0360  1             (bi,)
;  365        P 0361  1             (specified_bi,),
;  366        P 0362  1             (ru,),
```

```
;  367        P 0363  1          (specified_ru,),
;  368        P 0364  1          (only_ru,),
;  369        P 0365  1          (specified_only_ru,),
;  370        P 0366  1          (never_ru,),
;  371        P 0367  1          (specified_never_ru,),
;  372        P 0368  1          (ai_name,),
;  373        P 0369  1          (at_name,),
;  374        P 0370  1          (bi_name,),
;  375           0371  1          );
;  376           0372  1
;  377           0373  1 !
;  378           0374  1 ! Declare the miscellaneous flags
;  379           0375  1 !
;  380           0376  1 LITERAL
;  381        P 0377  1     $EQULST
;  382        P 0378  1          (MISC_,,1,1,
;  383        P 0379  1          (mark_file,),
;  384           0380  1          (already_rms$nlid,));
;  385           0381  1
;  386           0382  1 !
;  387           0383  1 ! Declare the error messages
;  388           0384  1 !
;  389           0385  1 !  Definitions in [CLIUTL.SRC]SET.MSG
;  390           0386  1 !
;  391           0387  1 EXTERNAL LITERAL
;  392           0388  1     lib$_quiconact,                 ! Quit asking for confirmation
;  393           0389  1     lib$_negans,                    ! Negative response to confirmation
;  394           0390  1     lib$_quipro,                    ! Quit processing
;  395           0391  1     lib$_filfaimat,                 ! Common qualifier match failed
;  396           0392  1     cli$_ivprot,                    ! Invalid protection
;  397           0393  1     cli$_negated,                   ! Value explicitly negated
;  398           0394  1     cli$_absent,                    ! Value not present and no defaul
;  399           0395  1     set$_operreq,                   ! OPER required
;  400           0396  1     set$_closeerr,                  ! Could not close file
;  401           0397  1     set$_entered,                   ! File entered in a directory
;  402           0398  1     set$_enterr,                    ! Error entering file
;  403           0399  1     set$_modified,                  ! File/directory modified
;  404           0400  1     set$_nonode,                    ! Node specification not allowed
;  405           0401  1     set$_notdir,                    ! Not a directory
;  406           0402  1     set$_notlocked,                 ! File not locked
;  407           0403  1     set$_notods2,                   ! Not an ODS2 structure
;  408           0404  1     set$_opendir,                   ! Could not open parent directory
;  409           0405  1     set$_pronotchg,                 ! Error message - "Protection not changed"
;  410           0406  1     set$_proerr,                    ! Error in changing the protection
;  411           0407  1     set$_protected,                 ! Informational log message for protection
;  412           0408  1     set$_readerr,                   ! Error reading the file
;  413           0409  1     set$_remerr,                    ! Could not remove file
;  414           0410  1     set$_removed,                   ! Directory entry removed
;  415           0411  1     set$_unlockerr,                 ! Could not lock file
;  416           0412  1     set$_writeerr,                  ! Error modifying file
;  417           0413  1     set$_unlocked;                  ! File unlocked
;  418           0414  1 !
;  419           0415  1 ! Define messages from the shared message facility
;  420           0416  1 !
;  421        P 0417  1 $SHR_MSGDEF (set, 119, global,
;  422        P 0418  1              (badlogic,  severe),     ! Fatal internal software error
;  423        P 0419  1              (badvalue,  error),      ! Invalid keyword value
```

```
;  424        P 0420  1       (valerr,    error),     ! Value out of range
;  425        P 0421  1       (syntax,    error),     ! Syntax problem
;  426        P 0422  1       (confqual,  error),     ! Conflicting qualifiers
;  427        P 0423  1       (delver,    error),     ! Explicit version number required
;  428        P 0424  1       (notrunc,   error),     ! Truncation not allowed
;  429        P 0425  1       (openin,    error),     ! Error opening a file
;  430          0426  1       (searchfail, error));   ! Error searching for a file
```

```
;  432     0427  1 GLOBAL ROUTINE set$file : NOVALUE =
;  433     0428  1
;  434     0429  1 !++
;  435     0430  1 !
;  436     0431  1 ! Functional description
;  437     0432  1 !
;  438     0433  1 !       This is the main control module.  It calls LIB$FILE_SCAN to perform
;  439     0434  1 !       the necessary functions on the file(s) specified in the call to SET.
;  440     0435  1 !
;  441     0436  1 ! Calling sequence
;  442     0437  1 !
;  443     0438  1 !       CALL set$file()
;  444     0439  1 !
;  445     0440  1 ! Input parameters
;  446     0441  1 !       none
;  447     0442  1 !
;  448     0443  1 ! Output parameters
;  449     0444  1 !       none
;  450     0445  1 !
;  451     0446  1 ! Implicit outputs
;  452     0447  1 !       none
;  453     0448  1 !
;  454     0449  1 ! Routine value
;  455     0450  1 !       none
;  456     0451  1 !
;  457     0452  1 ! Side effects
;  458     0453  1 !       none
;  459     0454  1 !
;  460     0455  1 !--
;  461     0456  1
;  462     0457  2 BEGIN
;  463     0458  2
;  464     0459  2 LOCAL
;  465     0460  2     desc : $BBLOCK[dsc$c_s_bln],
;  466     0461  2     scan_context,                   ! Sticky context argument for FILE$SCAN
;  467     0462  2     status;
;  468     0463  2
;  469     0464  2 !
;  470     0465  2 ! Check to make sure that the image is running with correct privilege.
;  471     0466  2 !
;  472     0467  2 check_privilege();
;  473     0468  2
;  474     0469  2 !
;  475     0470  2 ! Get the common qualifiers
;  476     0471  2 !
;  477     0472  2 status = lib$qual_file_parse(%REF(lib$m_cqf_exclude OR
;  478     0473  2                                   lib$m_cqf_before OR
;  479     0474  2                                   lib$m_cqf_since OR
;  480     0475  2                                   lib$m_cqf_created OR
;  481     0476  2                                   lib$m_cqf_modified OR
;  482     0477  2                                   lib$m_cqf_byowner),
;  483     0478  2                              context);
;  484     0479  2 IF NOT .status
;  485     0480  2 THEN (SIGNAL(.status); RETURN);
;  486     0481  2
;  487     0482  2 !
;  488     0483  2 ! Now to get all the command qualifiers.
```

```
2 !
2 IF NOT get_quals()
2 THEN RETURN;
2
2 !
2 ! Check to make sure that conflicting qualifiers were not specified.  If
2 ! they were, signal an error and stop.
2 !
2
2 IF .setfile$flags[qual_data]
2 THEN
2     IF .setfile$dflags[data_read] AND .setfile$dflags[data_noread]
2     OR .setfile$dflags[data_write] AND .setfile$dflags[data_nowrite]
2     THEN SIGNAL_STOP(set$_confqual);
2
2 IF .setfile$flags[qual_journal]
2 THEN
3     IF ( .setfile$jflags[jrnl_ru] AND .setfile$jflags[jrnl_only_ru] )
3     OR ( .setfile$jflags[jrnl_ru] AND .setfile$jflags[jrnl_never_ru] )
3     OR ( .setfile$jflags[jrnl_only_ru] AND .setfile$jflags[jrnl_never_ru] )
2     THEN SIGNAL_STOP(set$_confqual);
2
2 !
2 ! Next, for each file specified, find the file and perform the operations
2 ! requested.
2 !
2 scan_context = 0;                            ! Argument must be zero for file scan
2 $init_dyndesc(desc);                         ! Make a dynamic descriptor
2 WHILE cli$get_value(%ASCID 'FILE', desc)
2 AND NOT .setfile$flags[qual_quit] DO
3     BEGIN
3     file_fab[fab$b_fns] = .desc[dsc$w_length];
3     file_fab[fab$l_fna] = .desc[dsc$a_pointer];
3     lib$file_scan(                           ! For each file found,
3         file_fab,                            ! Use this fab
3         set_attributes,                      ! Go here if file found
3         search_error,                        ! Go here if error
3         scan_context)
3     END;
2
2 RETURN;
1 END;                                         ! End of routine set$file

                                                    .TITLE  SETFILE
                                                    .IDENT  \V04-000\

                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

                45  4C  49  46  00000 P.AAB:        .ASCII  \FILE\
                      010E0004  00004 P.AAA:        .LONG   17694724
                     00000000'  00008               .ADDRESS P.AAB

                                                    .PSECT  $OWN$,NOEXE,2

                                00000 CONTEXT:      .BLKB   4
```

```
                                .PSECT  $GLOBAL$,NOEXE,2

00000000  00000 SETFILE$FLAGS::
                                .LONG   0                               ;
00000000  00004 SETFILE$DFLAGS::
                                .LONG   0                               ;
00000000  00008 SETFILE$JFLAGS::
                                .LONG   0                               ;
00000000  0000C SETFILE$MFLAGS::
                                .LONG   0                               ;
     0000 00010 SETPRO_PROT::
                                .WORD   0                               ;
     0000 00012 SETPRO_MASK::
                                .WORD   0                               ;
          00014 GLOBAL_PROT::
                                .BLKB   2
          00016 GLOBAL_MASK::
                                .BLKB   2
          00018 EXP_VALUE::
                                .BLKB   8
          00020 EXTE_VALUE::
                                .BLKB   4
          00024 GBUF_VALUE::
                                .BLKB   4
          00028 UIC_VALUE::
                                .BLKB   4
          0002C GROUP::         .BLKB   4
          00030 MEMBER::        .BLKB   4
          00034 VRSN_VALUE::
                                .BLKB   4
          00038 RENAME_BUF::
                                .BLKB   255
          00137                 .BLKB   1
          00138 FILE_NAME::
                                .BLKB   8
          00140 AI_JNL_NAME::
                                .BLKB   16
          00150 AT_JNL_NAME::
                                .BLKB   16
          00160 BI_JNL_NAME::
                                .BLKB   16
       00# 00170 AI_JNL_DESC::
                                .BYTE   0[4]
00000000' 00174                 .ADDRESS AI_JNL_NAME                    ;
       00# 00178 AT_JNL_DESC::
                                .BYTE   0[4]
00000000' 0017C                 .ADDRESS AT_JNL_NAME                    ;
       00# 00180 BI_JNL_DESC::
                                .BYTE   0[4]
00000000' 00184                 .ADDRESS BI_JNL_NAME                    ;
00000001  00188 WORST_ERROR::
                                .LONG   1                               ;
          0018C CONF_DESC::
                                .BLKB   8
          00194 OLDPRIV::
                                .BLKB   8
       00# 0019C NEWPRIV::
```

```
                                    .BYTE   0[3]                              ;
          10  0019F                 .BYTE   16                                ;
              001A0                 .BLKB   4
              001A4 FILE_RESULT::
                                    .BLKB   255
              002A3                 .BLKB   1
              002A4 FILE_EXPANDED::
                                    .BLKB   255
              003A3                 .BLKB   1
          02  003A4 FILE_NAM::
                                    .BYTE   2                                 ;
          60  003A5                 .BYTE   96                                ;
          FF  003A6                 .BYTE   -1                                ;
          00  003A7                 .BYTE   0                                 ;
   00000000'  003A8                 .ADDRESS FILE_RESULT                      ;
          00  003AC                 .BYTE   0                                 ;
          00  003AD                 .BYTE   0                                 ;
          FF  003AE                 .BYTE   -1                                ;
          00  003AF                 .BYTE   0                                 ;
   00000000'  003B0                 .ADDRESS FILE_EXPANDED                    ;
    00000000  003B4                 .LONG   0                                 ;
       0000#  003B8                 .WORD   0[8]                              ;
       0000#  003C8                 .WORD   0[3]                              ;
       0000#  003CE                 .WORD   0[3]                              ;
    00000000  003D4                 .LONG   0                                 ;
    00000000  003D8                 .LONG   0                                 ;
          00  003DC                 .BYTE   0                                 ;
          00  003DD                 .BYTE   0                                 ;
          00  003DE                 .BYTE   0                                 ;
          00  003DF                 .BYTE   0                                 ;
          00  003E0                 .BYTE   0                                 ;
          00  003E1                 .BYTE   0                                 ;
         00#  003E2                 .BYTE   0[2]                              ;
    00000000  003E4                 .LONG   0                                 ;
    00000000  003E8                 .LONG   0                                 ;
    00000000  003EC                 .LONG   0                                 ;
    00000000  003F0                 .LONG   0                                 ;
    00000000  003F4                 .LONG   0                                 ;
    00000000  003F8                 .LONG   0                                 ;
   00000000#  003FC                 .LONG   0[2]                              ;
          03  00404 FILE_FAB::
                                    .BYTE   3                                 ;
          50  00405                 .BYTE   80                                ;
        0000  00406                 .WORD   0                                 ;
    00000000  00408                 .LONG   0                                 ;
    00000000  0040C                 .LONG   0                                 ;
    00000000  00410                 .LONG   0                                 ;
    00000000  00414                 .LONG   0                                 ;
        0000  00418                 .WORD   0                                 ;
          02  0041A                 .BYTE   2                                 ;
          00  0041B                 .BYTE   0                                 ;
    00000000  0041C                 .LONG   0                                 ;
          00  00420                 .BYTE   0                                 ;
          00  00421                 .BYTE   0                                 ;
          00  00422                 .BYTE   0                                 ;
          02  00423                 .BYTE   2                                 ;
    00000000  00424                 .LONG   0                                 ;
```

```
                         00000000  00428          .LONG    0                                  ;
                         00000000' 0042C          .ADDRESS FILE_NAM                           ;
                         00000000  00430          .LONG    0                                  ;
                         00000000  00434          .LONG    0                                  ;
                               00  00438          .BYTE    0                                  ;
                               00  00439          .BYTE    0                                  ;
                             0000  0043A          .WORD    0                                  ;
                         00000000  0043C          .LONG    0                                  ;
                             0000  00440          .WORD    0                                  ;
                               00  00442          .BYTE    0                                  ;
                               00  00443          .BYTE    0                                  ;
                         00000000  00444          .LONG    0                                  ;
                         00000000  00448          .LONG    0                                  ;
                             0000  0044C          .WORD    0                                  ;
                               00  0044E          .BYTE    0                                  ;
                               00  0044F          .BYTE    0                                  ;
                         00000000  00450          .LONG    0                                  ;

                                      SET$_BADLOGIC==      7803172
                                      SET$_BADVALUE==      7803154
                                      SET$_VALERR==        7803370
                                      SET$_SYNTAX==        7803130
                                      SET$_CONFQUAL==      7803618
                                      SET$_DELVER==        7803402
                                      SET$_NOTRUNC==       7803650
                                      SET$_OPENIN==        7803034
                                      SET$_SEARCHFAIL==    7803450
                                          .EXTRN   PARSE_UIC, CLI$PRESENT
                                          .EXTRN   CLI$GET_VALUE, LIB$CVT_DTB
                                          .EXTRN   LIB$CVT_TIME, LIB$FILE_SCAN
                                          .EXTRN   LIB$QUAL_FILE_PARSE
                                          .EXTRN   LIB$QUAL_FILE_MATCH
                                          .EXTRN   LIB$CONFIRM_ACT
                                          .EXTRN   LIB$GET_COMMAND
                                          .EXTRN   LIB$UNLOCK_FILE
                                          .EXTRN   SYS$FAO, SYS$SETPRV
                                          .EXTRN   LIB$SET_FILE_PROT
                                          .EXTRN   LIB$_QUICONACT, LIB$_NEGANS
                                          .EXTRN   LIB$_QUIPRO, LIB$_FICFAIMAT
                                          .EXTRN   CLI$_IVPROT, CLI$_NEGATED
                                          .EXTRN   CLI$_ABSENT, SET$_OPERREQ
                                          .EXTRN   SET$_CLOSEERR, SET$_ENTERED
                                          .EXTRN   SET$_ENTERR, SET$_MODIFIED
                                          .EXTRN   SET$_NONODE, SET$_NOTDIR
                                          .EXTRN   SET$_NOTLOCKED, SET$_NOTODS2
                                          .EXTRN   SET$_OPENDIR, SET$_PRONOTCHG
                                          .EXTRN   SET$_PROERR, SET$_PROTECTED
                                          .EXTRN   SET$_READERR, SET$_REMERR
                                          .EXTRN   SET$_REMOVED, SET$_UNLOCKERR
                                          .EXTRN   SET$_WRITEERR, SET$_UNLOCKED

                                          .PSECT   $CODE$,NOWRT,2

                             000C  00000  .ENTRY   SET$FILE, Save R2,R3               ; 0427
53 00000000G   00   9E  00002            MOVAB    LIB$STOP, R3                        ;
52 00000000'   EF   9E  00009            MOVAB    SETFILE$JFLAGS, R2                  ;
5E             10   C2  00010            SUBL2    #16, SP                             ;
```

```
                00000000V  EF                00  FB 00013          CALLS   #0, CHECK_PRIVILEGE                      ;0467
                        00000000'  EF     9F  00  9F 0001A          PUSHAB  CONTEXT                                 ;0472
                              04   AE   013E  8F  3C 00020          MOVZWL  #318, 4(SP)                             ;0476
                                04  AE      9F  00026              PUSHAB  4(SP)                                   ;0472
                00000000G  00                02  FB 00029          CALLS   #2, LIB$QUAL_FILE_PARSE                 ;
                           0A                50  E8 00030          BLBS    STATUS, 1$                              ;0479
                                             50  DD 00033          PUSHL   STATUS                                  ;0480
                00000000G  00                01  FB 00035          CALLS   #1, LIB$SIGNAL                          ;
                                             04  00 0003C          RET                                             ;
                00000000V  EF                00  FB 0003D  1$:     CALLS   #0, GET_QUALS                           ;0485
                           68                50  E9 00044          BLBC    R0, 9$                                  ;
                1D         F8   A2           04  E1 00047          BBC     #4, SETFILE$FLAGS, 4$                   ;0493
                05         FC   A2           01  E1 0004C          BBC     #1, SETFILE$DFLAGS, 2$                  ;0495
                0A         FC   A2           03  E0 00051          BBS     #3, SETFILE$DFLAGS, 3$                  ;
                0E         FC   A2           02  E1 00056  2$:     BBC     #2, SETFILE$DFLAGS, 4$                  ;0496
                09         FC   A2           04  E1 0005B          BBC     #4, SETFILE$DFLAGS, 4$                  ;
                                   007712E2    8F  DD 00060  3$:     PUSHL   #7803618                                ;0497
                              63               01  FB 00066          CALLS   #1, LIB$STOP                            ;
                23         F9   A2           03  E1 00069  4$:     BBC     #3, SETFILE$FLAGS+1, 7$                 ;0499
                                             62  95 0006E          TSTB    SETFILE$JFLAGS                          ;0501
                                             0C  18 00070          BGEQ    5$                                      ;
                11         01   A2           01  E0 00072          BBS     #1, SETFILE$JFLAGS+1, 6$                ;
                                             05  18 00077          BGEQ    5$                                      ;0502
                0A         01   A2           03  E0 00079          BBS     #3, SETFILE$JFLAGS+1, 6$                ;
                0E         01   A2           01  E1 0007E  5$:     BBC     #1, SETFILE$JFLAGS+1, 7$                ;0503
                09         01   A2           03  E1 00083          BBC     #3, SETFILE$JFLAGS+1, 7$                ;
                                   007712E2    8F  DD 00088  6$:     PUSHL   #7803618                                ;0504
                              63               01  FB 0008E          CALLS   #1, LIB$STOP                            ;
                                    04     AE  D4  00091  7$:     CLRL    SCAN_CONTEXT                            ;0510
                              08   AE 020E0000  8F  D0 00094          MOVL    #34471936, DESC                         ;0511
                                       0C     AE  D4  0009C          CLRL    DESC+4                                  ;
                                       08     AE  9F  0009F  8$:     PUSHAB  DESC                                    ;0512
                                00000000'     EF  9F  000A2          PUSHAB  P.AAA                                   ;
                00000000G  00                02  FB 000A8          CALLS   #2, CLI$GET_VALUE                       ;
                           2D                50  E9 000AF  9$:     BLBC    R0, 10$                                 ;
                28         FA   A2           06  E0 000B2          BBS     #6, SETFILE$FLAGS+2, 10$                ;0513
                         0430   C2     08     AE  90 000B7          MOVB    DESC, FILE_FAB+52                       ;0515
                         0428   C2     0C     AE  D0 000BD          MOVL    DESC+4, FILE_FAB+44                     ;0516
                                       04     AE  9F 000C3          PUSHAB  SCAN_CONTEXT                            ;0517
                                00000000V     EF  9F 000C6          PUSHAB  SEARCH_ERROR                            ;
                                00000000V     EF  9F 000CC          PUSHAB  SET_ATTRIBUTES                          ;
                                        03FC  C2  9F 000D2          PUSHAB  FILE_FAB                                ;
                00000000G  00                04  FB 000D6          CALLS   #4, LIB$FILE_SCAN                       ;
                                             C0  11 000DD          BRB     8$                                      ;
                                             04  00 000DF  10$:     RET                                             ;0525

; Routine Size:  224 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   532          0526  1 ROUTINE get_quals =
;   533          0527  1 !++
;   534          0528  1 !
;   535          0529  1 ! This routine gets all the qualifiers and values.
;   536          0530  1 !
;   537          0531  1 !
;   538          0532  2 BEGIN
;   539          0533  2
;   540          0534  2 LOCAL
;   541          0535  2     status,
;   542          0536  2     desc : $BBLOCK[dsc$c_s_bln];
;   543          0537  2
;   544          0538  2 $init_dyndesc(desc);                    ! Make a dynamic descriptor
;   545          0539  2
;   546          0540  2 !
;   547          0541  2 ! /[NO]BACKUP
;   548          0542  2 !
;   549          0543  2 status = cli$present(%ASCID 'BACKUP');
;   550          0544  2 IF .status
;   551          0545  2 THEN setfile$flags[qual_backup] = 1
;   552          0546  2 ELSE IF .status EQL cli$_negated
;   553          0547  2 THEN setfile$flags[qual_nobackup] = 1;
;   554          0548  2
;   555          0549  2 !
;   556          0550  2 ! /CONFIRM
;   557          0551  2 !
;   558          0552  2 IF cli$present(%ASCID 'CONFIRM')
;   559          0553  2 THEN setfile$flags[qual_confirm] = 1;
;   560          0554  2
;   561          0555  2 !
;   562          0556  2 ! /DATA_CHECK
;   563          0557  2 !
;   564          0558  2 IF cli$present(%ASCID 'DATA_CHECK')
;   565          0559  2 THEN
;   566          0560  3     BEGIN
;   567          0561  3     setfile$flags[qual_data] = 1;
;   568          0562  3     IF NOT cli$get_value(%ASCID 'DATA_CHECK', desc)
;   569          0563  3     THEN setfile$dflags[data_write] = 1
;   570          0564  3     ELSE INCR i FROM 0 TO 1 DO
;   571          0565  4         BEGIN
;   572          0566  4         IF CH$EQL(.desc[dsc$w_length], .desc[dsc$a_pointer],
;   573          0567  4                   .desc[dsc$w_length], UPLIT(BYTE('WRITE')))
;   574          0568  4         THEN setfile$dflags[data_write] = 1
;   575          0569  4         ELSE IF CH$EQL(.desc[dsc$w_length], .desc[dsc$a_pointer],
;   576          0570  4                   .desc[dsc$w_length], UPLIT(BYTE('READ')))
;   577          0571  4         THEN setfile$dflags[data_read] = 1
;   578          0572  4         ELSE
;   579          0573  5             BEGIN
;   580          0574  5             SIGNAL(set$_syntax, 1, desc);
;   581          0575  5             RETURN false;
;   582          0576  4             END;
;   583          0577  4         IF NOT cli$get_value(%ASCID 'DATA_CHECK', desc)
;   584          0578  4         THEN EXITLOOP
;   585          0579  3         END;
;   586          0580  2     END;
;   587          0581  2
;   588          0582  2 !
```

```
;  589     0583  2 ! /ENTER
;  590     0584  2 !
;  591     0585  2 IF cli$get_value(%ASCID 'ENTER', desc)
;  592     0586  2 THEN
;  593     0587  3     BEGIN
;  594     0588  3     setfile$flags[qual_enter] = 1;
;  595     0589  3     CH$MOVE(.desc[dsc$w_length],
;  596     0590  3             .desc[dsc$a_pointer],
;  597     0591  3             rename_buf);
;  598     0592  3     file_name[0] = .desc[dsc$w_length];
;  599     0593  3     file_name[1] = .desc[dsc$a_pointer];
;  600     0594  3     $init_dyndesc(desc);
;  601     0595  2     END;
;  602     0596  2
;  603     0597  2 !
;  604     0598  2 ! /END_OF_FILE
;  605     0599  2 !
;  606     0600  2 IF cli$present(%ASCID 'END_OF_FILE')
;  607     0601  2 THEN setfile$flags[qual_eof] = 1;
;  608     0602  2
;  609     0603  2 !
;  610     0604  2 ! /[NO]ERASE_ON_DELETE
;  611     0605  2 !
;  612     0606  2 status = cli$present(%ASCID 'ERASE_ON_DELETE');
;  613     0607  2 IF .status
;  614     0608  2 THEN setfile$flags[qual_erase] = 1
;  615     0609  2 ELSE IF .status EQL cli$_negated
;  616     0610  2 THEN setfile$flags[qual_noerase] = 1;
;  617     0611  2
;  618     0612  2 !
;  619     0613  2 ! /EXPIRATION_DATE
;  620     0614  2 !
;  621     0615  2 IF cli$present(%ASCID 'EXPIRATION_DATE')
;  622     0616  2 THEN
;  623     0617  2     IF cli$get_value(%ASCID 'EXPIRATION_DATE', desc)
;  624     0618  2     THEN
;  625     0619  3         BEGIN
;  626     0620  3         setfile$flags[qual_expi] = 1;
;  627     0621  3         IF NOT lib$cvt_time(desc, exp_value)
;  628     0622  3         THEN
;  629     0623  4             BEGIN
;  630     0624  4             SIGNAL(set$_syntax, 1, desc);
;  631     0625  4             RETURN false;
;  632     0626  3             END;
;  633     0627  2         END;
;  634     0628  2
;  635     0629  2 !
;  636     0630  2 ! /EXTENSION
;  637     0631  2 !
;  638     0632  2 IF cli$present(%ASCID 'EXTENSION')
;  639     0633  2 THEN
;  640     0634  3     BEGIN
;  641     0635  3     setfile$flags[qual_exte] = 1;
;  642     0636  3     exte_value = 5;
;  643     0637  3     IF cli$get_value(%ASCID 'EXTENSION', desc)
;  644     0638  3     THEN
;  645     0639  4         BEGIN
```

```
;  646          0640  4          IF NOT lib$cvt_dtb(.desc[dsc$w_length],
;  647          0641  4                             .desc[dsc$a_pointer],
;  648          0642  4                             exte_value)
;  649          0643  4          THEN
;  650          0644  5              BEGIN
;  651          0645  5              SIGNAL(set$_syntax, 1, desc);
;  652          0646  5              RETURN false;
;  653          0647  4              END;
;  654          0648  4          IF .exte_value LSS 0
;  655          0649  4          OR .exte_value GTR 65535
;  656          0650  4          THEN
;  657          0651  5              BEGIN
;  658          0652  5              SIGNAL(set$_syntax, 1, desc, set$_valerr);
;  659          0653  5              RETURN false;
;  660          0654  4              END;
;  661          0655  3          END;
;  662          0656  2      END;
;  663          0657  2
;  664          0658  2  !
;  665          0659  2  ! /GLOBAL_BUFFERS
;  666          0660  2  !
;  667          0661  2  IF cli$present(%ASCID 'GLOBAL_BUFFERS')
;  668          0662  2  THEN
;  669          0663  2      IF cli$get_value(%ASCID 'GLOBAL_BUFFERS', desc)
;  670          0664  2      THEN
;  671          0665  3          BEGIN
;  672          0666  3          setfile$flags[qual_gbuf] = 1;
;  673          0667  3          IF NOT lib$cvt_dtb(.desc[dsc$w_length],
;  674          0668  3                             .desc[dsc$a_pointer],
;  675          0669  3                             gbuf_value)
;  676          0670  3          THEN
;  677          0671  4              BEGIN
;  678          0672  4              SIGNAL(set$_syntax, 1, desc);
;  679          0673  4              RETURN false;
;  680          0674  3              END;
;  681          0675  3          IF .gbuf_value GTR 65535
;  682          0676  3          OR .gbuf_value LSS 0
;  683          0677  3          THEN
;  684          0678  4              BEGIN
;  685          0679  4              SIGNAL(set$_syntax, 1, desc, set$_valerr);
;  686          0680  4              RETURN false;
;  687          0681  3              END;
;  688          0682  2          END;
;  689          0683  2
;  690          0684  2  !
;  691          0685  2  ! /JOURNAL
;  692          0686  2  !
;  693          0687  3  begin                                  !**JNL**
;  694          0688  3  global set$gl_journaling;              !**JNL**
;  695          0689  3  if .set$gl_journaling                  !**JNL**
;  696          0690  3  then                                   !**JNL**
;  697          0691  3  IF cli$present(%ASCID 'JOURNAL')
;  698          0692  3  THEN
;  699          0693  4      BEGIN
;  700          0694  4      setfile$flags[qual_journal] = 1;
;  701          0695  4
;  702          0696  4      !
```

```
;  703          0697  4      ! /JOURNAL=AI=ai_journal_name
;  704          0698  4      !
;  705          0699  4      status = cli$present( %ASCID 'JOURNAL.AI', desc );
;  706          0700  4      IF .status NEQU cli$_absent
;  707          0701  4      THEN
;  708          0702  5          BEGIN
;  709          0703  5          setfile$jflags[jrnl_specified_ai] = 1;
;  710          0704  5          IF .status
;  711          0705  5          THEN
;  712          0706  5              setfile$jflags[jrnl_ai] = 1
;  713          0707  5          ELSE if .status EQLU cli$_negated
;  714          0708  5          THEN
;  715          0709  5              setfile$jflags[jrnl_ai] = 0;
;  716          0710  5          IF cli$get_value( %ASCID 'JOURNAL.AI', desc )
;  717          0711  5          THEN
;  718          0712  6              BEGIN
;  719          0713  6              IF .desc[dsc$w_length] GTRU cjf$c_mxjnlnaml
;  720          0714  6              THEN
;  721          0715  6                  SIGNAL( set$_badvalue, 1, desc );
;  722          0716  6              setfile$jflags[jrnl_ai_name] = 1;
;  723          0717  6              ai_jnl_desc[dsc$w_length] = .desc[dsc$w_length];
;  724          0718  6              CH$MOVE( .desc[dsc$w_length], .desc[dsc$a_pointer], ai_jnl_name );
;  725          0719  5              END;
;  726          0720  4          END;
;  727          0721  4
;  728          0722  4      !
;  729          0723  4      ! /JOURNAL=AT=at_journal_name
;  730          0724  4      !
;  731          0725  4      status = cli$present( %ASCID 'JOURNAL.AT', desc );
;  732          0726  4      IF .status NEQU cli$_absent
;  733          0727  4      THEN
;  734          0728  5          BEGIN
;  735          0729  5          setfile$jflags[jrnl_specified_at] = 1;
;  736          0730  5          IF .status
;  737          0731  5          THEN
;  738          0732  5              setfile$jflags[jrnl_at] = 1
;  739          0733  5          ELSE if .status EQLU cli$_negated
;  740          0734  5          THEN
;  741          0735  5              setfile$jflags[jrnl_at] = 0;
;  742          0736  5          IF cli$get_value( %ASCID 'JOURNAL.AT', desc )
;  743          0737  5          THEN
;  744          0738  6              BEGIN
;  745          0739  6              IF .desc[dsc$w_length] GTRU cjf$c_mxjnlnaml
;  746          0740  6              THEN
;  747          0741  6                  SIGNAL( set$_badvalue, 1, desc );
;  748          0742  6              setfile$jflags[jrnl_at_name] = 1;
;  749          0743  6              at_jnl_desc[dsc$w_length] = .desc[dsc$w_length];
;  750          0744  6              CH$MOVE( .desc[dsc$w_length], .desc[dsc$a_pointer], at_jnl_name );
;  751          0745  5              END;
;  752          0746  4          END;
;  753          0747  4
;  754          0748  4      !
;  755          0749  4      ! /JOURNAL=BI=bi_journal_name
;  756          0750  4      !
;  757          0751  4      status = cli$present( %ASCID 'JOURNAL.BI', desc );
;  758          0752  4      IF .status NEQU cli$_absent
;  759          0753  4      THEN
```

```
;  760     0754  5          BEGIN
;  761     0755  5          setfile$jflags[jrnl_specified_bi] = 1;
;  762     0756  5          IF .status
;  763     0757  5          THEN
;  764     0758  5              setfile$jflags[jrnl_bi] = 1
;  765     0759  5          ELSE if .status EQLU cli$_negated
;  766     0760  5          THEN
;  767     0761  5              setfile$jflags[jrnl_bi] = 0;
;  768     0762  5          IF cli$get_value( %ASCID 'JOURNAL.BI', desc )
;  769     0763  5          THEN
;  770     0764  6              BEGIN
;  771     0765  6              IF .desc[dsc$w_length] GTRU cjf$c_mxjnlnaml
;  772     0766  6              THEN
;  773     0767  6                  SIGNAL( set$_badvalue, 1, desc );
;  774     0768  6              setfile$jflags[jrnl_bi_name] = 1;
;  775     0769  6              bi_jnl_desc[dsc$w_length] = .desc[dsc$w_length];
;  776     0770  6              CH$MOVE( .desc[dsc$w_length], .desc[dsc$a_pointer], bi_jnl_name );
;  777     0771  5              END;
;  778     0772  4          END;
;  779     0773  4
;  780     0774  4      !
;  781     0775  4      ! /JOURNAL=RU
;  782     0776  4      !
;  783     0777  4      status = cli$present( %ASCID 'JOURNAL.RU', desc );
;  784     0778  4      IF .status NEQU cli$_absent
;  785     0779  4      THEN
;  786     0780  5          BEGIN
;  787     0781  5          setfile$jflags[jrnl_specified_ru] = 1;
;  788     0782  5          IF .status
;  789     0783  5          THEN
;  790     0784  5              setfile$jflags[jrnl_ru] = 1
;  791     0785  5          ELSE if .status EQLU cli$_negated
;  792     0786  5          THEN
;  793     0787  5              setfile$jflags[jrnl_ru] = 0;
;  794     0788  4          END;
;  795     0789  4
;  796     0790  4      !
;  797     0791  4      ! /JOURNAL=NEVER_RU
;  798     0792  4      !
;  799     0793  4      status = cli$present( %ASCID 'JOURNAL.NEVER_RU', desc );
;  800     0794  4      IF .status NEQU cli$_absent
;  801     0795  4      THEN
;  802     0796  5          BEGIN
;  803     0797  5          setfile$jflags[jrnl_specified_never_ru] = 1;
;  804     0798  5          IF .status
;  805     0799  5          THEN
;  806     0800  5              setfile$jflags[jrnl_never_ru] = 1
;  807     0801  5          ELSE if .status EQLU cli$_negated
;  808     0802  5          THEN
;  809     0803  5              setfile$jflags[jrnl_never_ru] = 0;
;  810     0804  4          END;
;  811     0805  4
;  812     0806  4      !
;  813     0807  4      ! /JOURNAL=ONLY_RU
;  814     0808  4      !
;  815     0809  4      status = cli$present( %ASCID 'JOURNAL.ONLY_RU', desc );
;  816     0810  4      IF .status NEQU cli$_absent
```

```
;  817          0811  4      THEN
;  818          0812  5          BEGIN
;  819          0813  5          setfile$jflags[jrnl_specified_only_ru] = 1;
;  820          0814  5          IF .status
;  821          0815  5          THEN
;  822          0816  5              setfile$jflags[jrnl_only_ru] = 1
;  823          0817  5          ELSE if .status EQLU cli$_negated
;  824          0818  5          THEN
;  825          0819  5              setfile$jflags[jrnl_only_ru] = 0;
;  826          0820  4          END;
;  827          0821  4
;  828          0822  3      END;
;  829          0823  2  end;     !**JNL**
;  830          0824  2
;  831          0825  2  !
;  832          0826  2  ! /LOG
;  833          0827  2  !
;  834          0828  2  setfile$flags[qual_log] = cli$present(%ASCID 'LOG');
;  835          0829  2
;  836          0830  2  !
;  837          0831  2  ! /NODIRECTORY
;  838          0832  2  !
;  839          0833  2  IF cli$present(%ASCID 'NODIRECTORY')
;  840          0834  2  THEN setfile$flags[qual_nodi] = 1;
;  841          0835  2
;  842          0836  2  !
;  843          0837  2  ! /OWNER_UIC
;  844          0838  2  !
;  845          0839  2  IF cli$present(%ASCID 'OWNER_UIC')
;  846          0840  2  THEN
;  847          0841  3      BEGIN
;  848          0842  3      setfile$flags[qual_owner] = 1;
;  849          0843  3      IF NOT cli$get_value(%ASCID 'OWNER_UIC', desc)
;  850          0844  3      THEN
;  851          0845  4          BEGIN
;  852          0846  4          LOCAL
;  853          0847  4              iosb : VECTOR[4,WORD];
;  854        P 0848  4          status = $GETJPIW(ITMLST = UPLIT(WORD(4,jpi$_uic),
;  855        P 0849  4                                           uic_value,
;  856        P 0850  4                                           0,
;  857        P 0851  4                                           0),
;  858          0852  4                                  IOSB = iosb);
;  859          0853  4          IF .status
;  860          0854  4          THEN status = .iosb[0];
;  861          0855  4          IF NOT .status
;  862          0856  4          THEN
;  863          0857  5              BEGIN
;  864          0858  5              SIGNAL(.status);
;  865          0859  5              RETURN false;
;  866          0860  4              END;
;  867          0861  4          END
;  868          0862  3      ELSE
;  869          0863  4          BEGIN
;  870          0864  4          IF CHSEQL(.desc[dsc$w_length], .desc[dsc$a_pointer],
;  871          0865  4                    .desc[dsc$w_length], UPLIT(BYTE('PARENT')))
;  872          0866  4          THEN setfile$flags[qual_parent] = 1
;  873          0867  5          ELSE IF NOT (status = parse_uic(desc, uic_value))
```

```
;  874     0868  4                THEN
;  875     0869  5                    BEGIN
;  876     0870  5                    SIGNAL(.status);
;  877     0871  5                    RETURN false;
;  878     0872  4                    END;
;  879     0873  3            END;
;  880     0874  2        END;
;  881     0875  2    !
;  882     0876  2    ! /PROTECTION
;  883     0877  2    !
;  884     0878  2    IF cli$present(%ASCID 'PROTECTION')
;  885     0879  2    THEN
;  886     0880  3    BEGIN
;  887     0881  3    LOCAL
;  888     0882  3          prot_desc : $BBLOCK[dsc$c_s_bln]; ! Protection descriptor
;  889     0883  3
;  890     0884  3      setfile$flags[qual_protection] = 1;
;  891     0885  3
;  892     0886  3    !
;  893     0887  3    !       Parse the /PROTECTION= value
;  894     0888  3    !
;  895     0889  3      $init_dyndesc(prot_desc);              ! Make a dynamic descriptor
;  896     0890  3
;  897     0891  3      IF cli$present(%ASCID 'PROTECTION.SYSTEM')
;  898     0892  3      THEN
;  899     0893  4          BEGIN
;  900     0894  4          setpro_mask = .setpro_mask OR %X'000F';
;  901     0895  4          IF cli$get_value(%ASCID 'PROTECTION.SYSTEM',prot_desc)
;  902     0896  4          THEN setpro_prot = parse_class(prot_desc);
;  903     0897  3          END;
;  904     0898  3      IF cli$present(%ASCID 'PROTECTION.OWNER')
;  905     0899  3      THEN
;  906     0900  4          BEGIN
;  907     0901  4          setpro_mask = .setpro_mask OR %X'00F0';
;  908     0902  4          IF cli$get_value(%ASCID 'PROTECTION.OWNER',prot_desc)
;  909     0903  4          THEN setpro_prot = .setpro_prot OR parse_class(prot_desc)^4;
;  910     0904  3          END;
;  911     0905  3      IF cli$present(%ASCID 'PROTECTION.GROUP')
;  912     0906  3      THEN
;  913     0907  4          BEGIN
;  914     0908  4          setpro_mask = .setpro_mask OR %X'0F00';
;  915     0909  4          IF cli$get_value(%ASCID 'PROTECTION.GROUP',prot_desc)
;  916     0910  4          THEN setpro_prot = .setpro_prot OR parse_class(prot_desc)^8;
;  917     0911  3          END;
;  918     0912  3      IF cli$present(%ASCID 'PROTECTION.WORLD')
;  919     0913  3      THEN
;  920     0914  4          BEGIN
;  921     0915  4          setpro_mask = .setpro_mask OR %X'F000';
;  922     0916  4          IF cli$get_value(%ASCID 'PROTECTION.WORLD',prot_desc)
;  923     0917  4          THEN setpro_prot = .setpro_prot OR parse_class(prot_desc)^12;
;  924     0918  3          END;
;  925     0919  3
;  926     0920  3    !
;  927     0921  3    ! Complement the protection value since at this point, a bit set true
;  928     0922  3    ! indicates that we want to ALLOW access, while the system convention
;  929     0923  3    ! is that a bit set true indicates that we want to DENY access.
;  930     0924  3    !
```

```
;  931     0925  3     IF .setpro_mask NEQ 0                   ! If any protections specified
;  932     0926  3     THEN setpro_prot = NOT .setpro_prot;    ! then get the complement
;  933     0927  3
;  934     0928  3     !
;  935     0929  3     ! Now save the command level protection in the protection
;  936     0930  3     ! area. If the user did not supply a command level protection then
;  937     0931  3     ! the global_mask will have a value of zero.
;  938     0932  3     !
;  939     0933  3     global_mask = .setpro_mask;
;  940     0934  3     global_prot = .setpro_prot;
;  941     0935  2     END;
;  942     0936  2     !
;  943     0937  2     ! /REMOVE
;  944     0938  2     !
;  945     0939  2     IF cli$present(%ASCID 'REMOVE')
;  946     0940  2     THEN setfile$flags[qual_remove] = 1;
;  947     0941  2
;  948     0942  2     !
;  949     0943  2     ! /TRUNCATE
;  950     0944  2     !
;  951     0945  2     IF cli$present(%ASCID 'TRUNCATE')
;  952     0946  2     THEN setfile$flags[qual_trunc] = 1;
;  953     0947  2
;  954     0948  2     !
;  955     0949  2     ! /UNLOCK
;  956     0950  2     !
;  957     0951  2     IF cli$present(%ASCID 'UNLOCK')
;  958     0952  2     THEN setfile$flags[qual_unlock] = 1;
;  959     0953  2
;  960     0954  2     !
;  961     0955  2     ! /VERSION_LIMIT
;  962     0956  2     !
;  963     0957  2     IF cli$present(%ASCID 'VERSION_LIMIT')
;  964     0958  2     THEN
;  965     0959  2         BEGIN
;  966     0960  3         setfile$flags[qual_vrsn] = 1;               ! Show that /VERSION specified
;  967     0961  3         vrsn_value = 32767;                         ! Set to the default
;  968     0962  3         IF cli$get_value(%ASCID 'VERSION_LIMIT', desc)
;  969     0963  3         THEN
;  970     0964  4             BEGIN
;  971     0965  4             IF NOT lib$cvt_dtb(.desc[dsc$w_length],
;  972     0966  4                                .desc[dsc$a_pointer],
;  973     0967  4                                vrsn_value)
;  974     0968  4             THEN
;  975     0969  5                 BEGIN
;  976     0970  5                 SIGNAL(set$_syntax, 1, desc);
;  977     0971  5                 RETURN false;
;  978     0972  4                 END;
;  979     0973  4
;  980     0974  4             IF .vrsn_value EQL 0
;  981     0975  4             THEN
;  982     0976  4                 vrsn_value = 32767;
;  983     0977  4
;  984     0978  4             IF .vrsn_value LSS 0
;  985     0979  4               OR .vrsn_value GTR 32767
;  986     0980  4             THEN
;  987     0981  4                 (SIGNAL(set$_valerr); RETURN false);
```

```
;  988          0982  3          END;
;  989          0983  2      END;
;  990          0984  2
;  991          0985  2  RETURN true;
;  992          0986  1  END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                    00  00  50  55  4B  43  41  42  0000C P.AAD:   .ASCII  \BACKUP\<0><0>
                                                        010E0006    00014 P.AAC:   .LONG   17694726
                                                        00000000'   00018          .ADDRESS P.AAD
                                    00  4D  52  49  46  4E  4F  43  0001C P.AAF:   .ASCII  \CONFIRM\<0>
                                                        010E0007    00024 P.AAE:   .LONG   17694727
                                                        00000000'   00028          .ADDRESS P.AAF
                00  00  4B  43  45  48  43  5F  41  54  41  44  0002C P.AAH:   .ASCII  \DATA_CHECK\<0><0>
                                                        010E000A    00038 P.AAG:   .LONG   17694730
                                                        00000000'   0003C          .ADDRESS P.AAH
                00  00  4B  43  45  48  43  5F  41  54  41  44  00040 P.AAJ:   .ASCII  \DATA_CHECK\<0><0>
                                                        010E000A    0004C P.AAI:   .LONG   17694730
                                                        00000000'   00050          .ADDRESS P.AAJ
                                            45  54  49  52  57  00054 P.AAK:   .ASCII  \WRITE\
                                                                    00059          .BLKB   3
                                        44  41  45  52              0005C P.AAL:   .ASCII  \READ\
                00  00  4B  43  45  48  43  5F  41  54  41  44  00060 P.AAN:   .ASCII  \DATA_CHECK\<0><0>
                                                        010E000A    0006C P.AAM:   .LONG   17694730
                                                        00000000'   00070          .ADDRESS P.AAN
                                    00  00  00  52  45  54  4E  45  00074 P.AAP:   .ASCII  \ENTER\<0><0><0>
                                                        010E0005    0007C P.AAO:   .LONG   17694725
                                                        00000000'   00080          .ADDRESS P.AAP
                        00  45  4C  49  46  5F  46  4F  5F  44  4E  45  00084 P.AAR:   .ASCII  \END_OF_FILE\<0>
                                                        010E0008    00090 P.AAQ:   .LONG   17694731
                                                        00000000'   00094          .ADDRESS P.AAR
45  54  45  4C  45  44  5F  4E  4F  5F  45  53  41  52  45  00098 P.AAT:   .ASCII  \ERASE_ON_DELETE\<0>
                                                            00      000A7
                                                        010E000F    000A8 P.AAS:   .LONG   17694735
                                                        00000000'   000AC          .ADDRESS P.AAT
45  54  41  44  5F  4E  4F  49  54  41  52  49  50  58  45  000B0 P.AAV:   .ASCII  \EXPIRATION_DATE\<0>
                                                            00      000BF
                                                        010E000F    000C0 P.AAU:   .LONG   17694735
                                                        00000000'   000C4          .ADDRESS P.AAV
45  54  41  44  5F  4E  4F  49  54  41  52  49  50  58  45  000C8 P.AAX:   .ASCII  \EXPIRATION_DATE\<0>
                                                            00      000D7
                                                        010E000F    000D8 P.AAW:   .LONG   17694735
                                                        00000000'   000DC          .ADDRESS P.AAX
                        00  00  00  4E  4F  49  53  4E  45  54  58  45  000E0 P.AAZ:   .ASCII  \EXTENSION\<0><0><0>
                                                        010E0009    000EC P.AAY:   .LONG   17694729
                                                        00000000'   000F0          .ADDRESS P.AAZ
                        00  00  00  4E  4F  49  53  4E  45  54  58  45  000F4 P.ABB:   .ASCII  \EXTENSION\<0><0><0>
                                                        010E0009    00100 P.ABA:   .LONG   17694729
                                                        00000000'   00104          .ADDRESS P.ABB
00  53  52  45  46  46  55  42  5F  4C  41  42  4F  4C  47  00108 P.ABD:   .ASCII  \GLOBAL_BUFFERS\<0><0>
                                                            00      00117
                                                        010E000E    00118 P.ABC:   .LONG   17694734
                                                        00000000'   0011C          .ADDRESS P.ABD
00  53  52  45  46  46  55  42  5F  4C  41  42  4F  4C  47  00120 P.ABF:   .ASCII  \GLOBAL_BUFFERS\<0><0>
```

```
                                                                00  0012F
                                                          010E000E  00130 P.ABE:   .LONG    17694734
                                                          00000000' 00134         .ADDRESS P.ABF
                                                    00 4C 41 4E 52 55 4F 4A 00138 P.ABH:   .ASCII   \JOURNAL\<0>
                                                          010E0007  00140 P.ABG:   .LONG    17694727
                                                          00000000' 00144         .ADDRESS P.ABH
                                              00 00 49 41 2E 4C 41 4E 52 55 4F 4A 00148 P.ABJ:   .ASCII   \JOURNAL.AI\<0><0>
                                                          010E000A  00154 P.ABI:   .LONG    17694730
                                                          00000000' 00158         .ADDRESS P.ABJ
                                              00 00 49 41 2E 4C 41 4E 52 55 4F 4A 0015C P.ABL:   .ASCII   \JOURNAL.AI\<0><0>
                                                          010E000A  00168 P.ABK:   .LONG    17694730
                                                          00000000' 0016C         .ADDRESS P.ABL
                                              00 00 54 41 2E 4C 41 4E 52 55 4F 4A 00170 P.ABN:   .ASCII   \JOURNAL.AT\<0><0>
                                                          010E000A  0017C P.ABM:   .LONG    17694730
                                                          00000000' 00180         .ADDRESS P.ABN
                                              00 00 54 41 2E 4C 41 4E 52 55 4F 4A 00184 P.ABP:   .ASCII   \JOURNAL.AT\<0><0>
                                                          010E000A  00190 P.ABO:   .LONG    17694730
                                                          00000000' 00194         .ADDRESS P.ABP
                                              00 00 49 42 2E 4C 41 4E 52 55 4F 4A 00198 P.ABR:   .ASCII   \JOURNAL.BI\<0><0>
                                                          010E000A  001A4 P.ABQ:   .LONG    17694730
                                                          00000000' 001A8         .ADDRESS P.ABR
                                              00 00 49 42 2E 4C 41 4E 52 55 4F 4A 001AC P.ABT:   .ASCII   \JOURNAL.BI\<0><0>
                                                          010E000A  001B8 P.ABS:   .LONG    17694730
                                                          00000000' 001BC         .ADDRESS P.ABT
                                              00 00 55 52 2E 4C 41 4E 52 55 4F 4A 001C0 P.ABV:   .ASCII   \JOURNAL.RU\<0><0>
                                                          010E000A  001CC P.ABU:   .LONG    17694730
                                                          00000000' 001D0         .ADDRESS P.ABV
52 5F 52 45 56 45 4E 2E 4C 41 4E 52 55 4F 4A 001D4 P.ABX:   .ASCII   \JOURNAL.NEVER_RU\
                                                                55  001E3
                                                          010E0010  001E4 P.ABW:   .LONG    17694736
                                                          00000000' 001E8         .ADDRESS P.ABX
55 52 5F 59 4C 4E 4F 2E 4C 41 4E 52 55 4F 4A 001EC P.ABZ:   .ASCII   \JOURNAL.ONLY_RU\<0>
                                                                00  001FB
                                                          010E000F  001FC P.ABY:   .LONG    17694735
                                                          00000000' 00200         .ADDRESS P.ABZ
                                                       00 47 4F 4C  00204 P.ACB:   .ASCII   \LOG\<0>
                                                          010E0003  00208 P.ACA:   .LONG    17694723
                                                          00000000' 0020C         .ADDRESS P.ACB
                                           00 59 52 4F 54 43 45 52 49 44 4F 4E 00210 P.ACD:   .ASCII   \NODIRECTORY\<0>
                                                          010E000B  0021C P.ACC:   .LONG    17694731
                                                          00000000' 00220         .ADDRESS P.ACD
                                           00 00 00 43 49 55 5F 52 45 4E 57 4F 00224 P.ACF:   .ASCII   \OWNER_UIC\<0><0><0>
                                                          010E0009  00230 P.ACE:   .LONG    17694729
                                                          00000000' 00234         .ADDRESS P.ACF
                                           00 00 00 43 49 55 5F 52 45 4E 57 4F 00238 P.ACH:   .ASCII   \OWNER_UIC\<0><0><0>
                                                          010E0009  00244 P.ACG:   .LONG    17694729
                                                          00000000' 00248         .ADDRESS P.ACH
                                                          0304 0004 0024C P.ACI:   .WORD    4, 772
                                                          00000000' 00250         .ADDRESS UIC_VALUE
                                                 00000000 00000000  00254         .LONG    0, 0
                                                    54 4E 45 52 41 50 0025C P.ACJ:   .ASCII   \PARENT\
                                                                      00262         .BLKB    2
                                           00 00 4E 4F 49 54 43 45 54 4F 52 50 00264 P.ACL:   .ASCII   \PROTECTION\<0><0>
                                                          010E000A  00270 P.ACK:   .LONG    17694730
                                                          00000000' 00274         .ADDRESS P.ACL
54 53 59 53 2E 4E 4F 49 54 43 45 54 4F 52 50 00278 P.ACN:   .ASCII   \PROTECTION.SYSTEM\<0><0><0>
                                                       00 00 00 4D 45  00287
```

```
                                                010E0011  002BC P.ACM:  .LONG   17694737
                                                00000000'  00290       .ADDRESS P.ACN
54  53  59  53  2E  4E  4F  49  54  43  45  54  4F  52  50  00294 P.ACP:  .ASCII  \PROTECTION.SYSTEM\<0><0><0>
                                        00  00  00  4D  45  002A3
                                                010E0011  002A8 P.ACO:  .LONG   17694737
                                                00000000'  002AC       .ADDRESS P.ACP
45  4E  57  4F  2E  4E  4F  49  54  43  45  54  4F  52  50  002B0 P.ACR:  .ASCII  \PROTECTION.OWNER\
                                                    52  002BF
                                                010E0010  002C0 P.ACQ:  .LONG   17694736
                                                00000000'  002C4       .ADDRESS P.ACR
45  4E  57  4F  2E  4E  4F  49  54  43  45  54  4F  52  50  002C8 P.ACT:  .ASCII  \PROTECTION.OWNER\
                                                    52  002D7
                                                010E0010  002D8 P.ACS:  .LONG   17694736
                                                00000000'  002DC       .ADDRESS P.ACT
55  4F  52  47  2E  4E  4F  49  54  43  45  54  4F  52  50  002E0 P.ACV:  .ASCII  \PROTECTION.GROUP\
                                                    50  002EF
                                                010E0010  002F0 P.ACU:  .LONG   17694736
                                                00000000'  002F4       .ADDRESS P.ACV
55  4F  52  47  2E  4E  4F  49  54  43  45  54  4F  52  50  002F8 P.ACX:  .ASCII  \PROTECTION.GROUP\
                                                    50  00307
                                                010E0010  00308 P.ACW:  .LONG   17694736
                                                00000000'  0030C       .ADDRESS P.ACX
4C  52  4F  57  2E  4E  4F  49  54  43  45  54  4F  52  50  00310 P.ACZ:  .ASCII  \PROTECTION.WORLD\
                                                    44  0031F
                                                010E0010  00320 P.ACY:  .LONG   17694736
                                                00000000'  00324       .ADDRESS P.ACZ
4C  52  4F  57  2E  4E  4F  49  54  43  45  54  4F  52  50  00328 P.ADB:  .ASCII  \PROTECTION.WORLD\
                                                    44  00337
                                                010E0010  00338 P.ADA:  .LONG   17694736
                                                00000000'  0033C       .ADDRESS P.ADB
                                    00  00  45  56  4F  4D  45  52  00340 P.ADD:  .ASCII  \REMOVE\<0><0>
                                                010E0006  00348 P.ADC:  .LONG   17694726
                                                00000000'  0034C       .ADDRESS P.ADD
                                    45  54  41  43  4E  55  52  54  00350 P.ADF:  .ASCII  \TRUNCATE\
                                                010E0008  00358 P.ADE:  .LONG   17694728
                                                00000000'  0035C       .ADDRESS P.ADF
                                    00  00  4B  43  4F  4C  4E  55  00360 P.ADH:  .ASCII  \UNLOCK\<0><0>
                                                010E0006  00368 P.ADG:  .LONG   17694726
                                                00000000'  0036C       .ADDRESS P.ADH
00  00  54  49  4D  49  4C  5F  4E  4F  49  53  52  45  56  00370 P.ADJ:  .ASCII  \VERSION_LIMIT\<0><0><0>
                                                    00  0037F
                                                010E000D  00380 P.ADI:  .LONG   17694733
                                                00000000'  00384       .ADDRESS P.ADJ
00  0C  54  49  4D  49  4C  5F  4E  4F  49  53  52  45  56  00388 P.ADL:  .ASCII  \VERSION_LIMIT\<0><0><0>
                                                    00  00397
                                                010E000D  00398 P.ADK:  .LONG   17694733
                                                00000000'  0039C       .ADDRESS P.ADL

                                                                       .PSECT  $GLOBAL$,NOEXE,2

                                                00454 SET$GL_JOURNALING::
                                                                       .BLKB   4

                                                                       .EXTRN  SYS$GETJPIW

                                                                       .PSECT  $CODE$,NOWRT,2
```

```
                                0FFC 00000 GET_QUALS:
                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11            ; 0526
                5B 00000000G   8F   D0 00002           MOVL    #CLI$_NEGATED, R11
                5A 00000000G   00   9E 00009           MOVAB   CLI$GET_VALUE, R10
                59 00000000G   00   9E 00010           MOVAB   CLI$PRESENT, R9
                58 00000000'   EF   9E 00017           MOVAB   P.AAC, R8
                57 00000000'   EF   9E 0001E           MOVAB   SETFILE$FLAGS, R7
                5E             10   C2 00025           SUBL2   #16, SP
             08 AE 020E0000    8F   D0 00028           MOVL    #34471936, DESC                         ; 0538
                         0C    AE   D4 00030           CLRL    DESC+4
                               58   DD 00033           PUSHL   R8                                      ; 0543
                            69 01   FB 00035           CALLS   #1, CLI$PRESENT
                            56 50   D0 00038           MOVL    R0, STATUS
                            05 56   E9 0003B           BLBC    STATUS, 1$                              ; 0544
                            67 02   88 0003E           BISB2   #2, SETFILE$FLAGS                       ; 0545
                               08   11 00041           BRB     2$
                            58 56   D1 00043 1$:       CMPL    STATUS, R11                             ; 0546
                               03   12 00046           BNEQ    2$
                            67 04   88 00048           BISB2   #4, SETFILE$FLAGS                       ; 0547
                         10    AB   9F 0004B 2$:       PUSHAB  P.AAE                                   ; 0552
                            69 01   FB 0004E           CALLS   #1, CLI$PRESENT
                            03 50   E9 00051           BLBC    R0, 3$
                            67 08   88 00054           BISB2   #8, SETFILE$FLAGS                       ; 0553
                         24    AB   9F 00057 3$:       PUSHAB  P.AAG                                   ; 0558
                            69 01   FB 0005A           CALLS   #1, CLI$PRESENT
                            46 50   E9 0005D           BLBC    R0, 9$
                            67 10   88 00060           BISB2   #16, SETFILE$FLAGS                      ; 0561
                         08    AE   9F 00063           PUSHAB  DESC                                    ; 0562
                         38    AB   9F 00066           PUSHAB  P.AAI
                            6A 02   FB 00069           CALLS   #2, CLI$GET_VALUE
                            06 50   E8 0006C           BLBS    R0, 4$
                      04    A7      04   88 0006F           BISB2   #4, SETFILE$DFLAGS                  ; 0563
                                    31   11 00073           BRB     9$
                                    54   D4 00075 4$:       CLRL    I                                   ; 0564
40 AB         0C    BE        08    AE   29 00077 5$:       CMPC3   DESC, @DESC+4, P.AAK                ; 0566
                                    06   12 0007E           BNEQ    6$
                      04    A7      04   88 00080           BISB2   #4, SETFILE$DFLAGS                  ; 0568
                                    10   11 00084           BRB     8$
48 AB         0C    BE        08    AE   29 00086 6$:       CMPC3   DESC, @DESC+4, P.AAL                ; 0569
                                    03   13 0008D           BEQL    7$
                                 04B1   31 0008F           BRW     56$
                      04    A7      02   88 00092 7$:       BISB2   #2, SETFILE$DFLAGS                  ; 0571
                                 08    AE   9F 00096 8$:       PUSHAB  DESC                            ; 0577
                                 58    AB   9F 00099           PUSHAB  P.AAM
                                    6A 02   FB 0009C           CALLS   #2, CLI$GET_VALUE
                                    04 50   E9 0009F           BLBC    R0, 9$
                   D1                54   01   F3 000A2           AOBLEQ  #1, I, 5$
                                 08    AE   9F 000A6 9$:       PUSHAB  DESC                            ; 0585
                                 68    AB   9F 000A9           PUSHAB  P.AAO
                                    6A 02   FB 000AC           CALLS   #2, CLI$GET_VALUE
                                    22 50   E9 000AF           BLBC    R0, 10$
                      02    A7      10   88 000B2           BISB2   #16, SETFILE$FLAGS+2                ; 0588
38 A7         0C    BE        08    AE   28 000B6           MOVC3   DESC, @DESC+4, RENAME_BUF           ; 0589
                0138  C7        08    AE   3C 000BD           MOVZWL  DESC, FILE_NAME                     ; 0592
                013C  C7        0C    AE   D0 000C3           MOVL    DESC+4, FILE_NAME+4                 ; 0593
                   08    AE 020E0000    8F   D0 000C9           MOVL    #34471936, DESC                     ; 0594
                                  0C    AE   D4 000D1           CLRL    DESC+4
```

```
                        7C  A8  9F  000D4 10$:    PUSHAB  P.AAQ                           ; 0600
                    69      01  FB  000D7        CALLS   #1, CLI$PRESENT
                    03      50  E9  000DA        BLBC    R0, 11$
                    67      20  88  000DD        BISB2   #32, SETFILE$FLAGS              ; 0601
                  0094      C8  9F  000E0 11$:    PUSHAB  P.AAS                           ; 0606
                    69      01  FB  000E4        CALLS   #1, CLI$PRESENT
                    56      50  D0  000E7        MOVL    R0, STATUS
                    06      56  E9  000EA        BLBC    STATUS, 12$                     ; 0607
                    67      40  8F  88  000ED    BISB2   #64, SETFILE$FLAGS              ; 0608
                            09  11  000F1        BRB     13$
                    5B      56  D1  000F3 12$:    CMPL    STATUS, R11                     ; 0609
                            04  12  000F6        BNEQ    13$
                    67      80  8F  88  000F8    BISB2   #128, SETFILE$FLAGS             ; 0610
                  00AC      C8  9F  000FC 13$:    PUSHAB  P.AAU                           ; 0615
                    69      01  FB  00100        CALLS   #1, CLI$PRESENT
                    21      50  E9  00103        BLBC    R0, 14$
                        08  AE  9F  00106        PUSHAB  DESC                            ; 0617
                  00C4      C8  9F  00109        PUSHAB  P.AAW
                    6A      02  FB  0010D        CALLS   #2, CLI$GET_VALUE
                    14      50  E9  00110        BLBC    R0, 14$
                01  A7      01  88  00113        BISB2   #1, SETFILE$FLAGS+1             ; 0620
                        18  A7  9F  00117        PUSHAB  EXP_VALUE                       ; 0621
                        0C  AE  9F  0011A        PUSHAB  DESC
            00000000G   00      02  FB  0011D    CALLS   #2, LIB$CVT_TIME
                    6E      50  E9  00124        BLBC    R0, 16$
                  00D8      C8  9F  00127 14$:    PUSHAB  P.AAY                           ; 0632
                    69      01  FB  0012B        CALLS   #1, CLI$PRESENT
                    38      50  E9  0012E        BLBC    R0, 15$
                01  A7      02  88  00131        BISB2   #2, SETFILE$FLAGS+1             ; 0635
                20  A7      05  D0  00135        MOVL    #5, EXTE_VALUE                  ; 0636
                        08  AE  9F  00139        PUSHAB  DESC                            ; 0637
                  00EC      C8  9F  0013C        PUSHAB  P.ABA
                    6A      02  FB  00140        CALLS   #2, CLI$GET_VALUE
                    23      50  E9  00143        BLBC    R0, 15$
                        20  A7  9F  00146        PUSHAB  EXTE_VALUE                      ; 0640
                        10  AE  DD  00149        PUSHL   DESC+4                          ; 0641
                    7E  10  AE  3C  0014C        MOVZWL  DESC, -(SP)                     ; 0640
            00000000G   00      03  FB  00150    CALLS   #3, LIB$CVT_DTB
                    3B      50  E9  00157        BLBC    R0, 16$
                    50  20  A7  D0  0015A        MOVL    EXTE_VALUE, R0                  ; 0648
                            4A  19  0015E        BLSS    18$
            0000FFFF    8F      50  D1  00160    CMPL    R0, #65535                      ; 0649
                            41  14  00167        BGTR    18$
                  0104      C8  9F  00169 15$:    PUSHAB  P.ABC                           ; 0661
                    69      01  FB  0016D        CALLS   #1, CLI$PRESENT
                    52      50  E9  00170        BLBC    R0, 19$
                        08  AE  9F  00173        PUSHAB  DESC                            ; 0663
                  011C      C8  9F  00176        PUSHAB  P.ABE
                    6A      02  FB  0017A        CALLS   #2, CLI$GET_VALUE
                    45      50  E9  0017D        BLBC    R0, 19$
                01  A7      04  88  00180        BISB2   #4, SETFILE$FLAGS+1             ; 0666
                        24  A7  9F  00184        PUSHAB  GBUF_VALUE                      ; 0667
                        10  AE  DD  00187        PUSHL   DESC+4                          ; 0668
                    7E  10  AE  3C  0018A        MOVZWL  DESC, -(SP)                     ; 0667
            00000000G   00      03  FB  0018E    CALLS   #3, LIB$CVT_DTB
                    03      50  E8  00195 16$:    BLBS    R0, 17$
                          03A8  31  00198        BRW     56$
```

```
                0000FFFF  8F        24  A7  D1 0019B 17$:    CMPL    GBUF_VALUE, #65535                ; 0675
                                        05  14 001A3         BGTR    18$
                                    24  A7  D5 001A5         TSTL    GBUF_VALUE                        ; 0676
                                        1B  18 001A8         BGEQ    19$
                                007711EA  8F  DD 001AA 18$:  PUSHL   #7803370                          ; 0679
                                    0C  AE  9F 001B0         PUSHAB  DESC
                                        01  DD 001B3         PUSHL   #1
                                007710FA  8F  DD 001B5       PUSHL   #7803130
                00000000G  00           04  FB 001BB         CALLS   #4, LIB$SIGNAL
                                      03BF  31 001C2         BRW     62$                               ; 0680
                           03     0454  C7  E8 001C5 19$:    BLBS    SET$GL_JOURNALING, 21$            ; 0689
                                      01C0  31 001CA 20$:    BRW     39$
                                  012C  C8  9F 001CD 21$:    PUSHAB  P.ABG                             ; 0691
                           69           01  FB 001D1         CALLS   #1, CLI$PRESENT
                           F3           50  E9 001D4         BLBC    R0, 20$
                      01   A7           08  88 001D7         BISB2   #8, SETFILE$FLAGS+1               ; 0694
                                    08  AE  9F 001DB         PUSHAB  DESC                              ; 0699
                                  0140  C8  9F 001DE         PUSHAB  P.ABI
                           69           02  FB 001E2         CALLS   #2, CLI$PRESENT
                           56           50  D0 001E5         MOVL    R0, STATUS
                00000000G  8F           56  D1 001E8         CMPL    STATUS, #CLI$_ABSENT              ; 0700
                                        4D  13 001EF         BEQL    25$
                      08   A7           04  88 001F1         BISB2   #4, SETFILE$JFLAGS                ; 0703
                           06           56  E9 001F5         BLBC    STATUS, 22$                       ; 0704
                      08   A7           02  88 001F8         BISB2   #2, SETFILE$JFLAGS                ; 0706
                                        09  11 001FC         BRB     23$
                           5B           56  D1 001FE 22$:    CMPL    STATUS, R11                       ; 0707
                                        04  12 00201         BNEQ    23$
                      08   A7           02  8A 00203         BICB2   #2, SETFILE$JFLAGS                ; 0709
                                    08  AE  9F 00207 23$:    PUSHAB  DESC                              ; 0710
                                  0154  C8  9F 0020A         PUSHAB  P.ABK
                           6A           02  FB 0020E         CALLS   #2, CLI$GET_VALUE
                           2A           50  E9 00211         BLBC    R0, 25$
                           10       08  AE  B1 00214         CMPW    DESC, #16                         ; 0713
                                        12  1B 00218         BLEQU   24$
                                    08  AE  9F 0021A         PUSHAB  DESC                              ; 0715
                                        01  DD 0021D         PUSHL   #1
                                00771112  8F  DD 0021F       PUSHL   #7803154
                00000000G  00           03  FB 00225         CALLS   #3, LIB$SIGNAL
                       09  A7           20  88 0022C 24$:    BISB2   #32, SETFILE$JFLAGS+1             ; 0716
                     0170  C7       08  AE  B0 00230         MOVW    DESC, AI_JNL_DESC                 ; 0717
      0140    C7      0C   BE       08  AE  28 00236         MOVC3   DESC, @DESC+4, AI_JNL_NAME        ; 0718
                                    08  AE  9F 0023E 25$:    PUSHAB  DESC                              ; 0725
                                  0168  C8  9F 00241         PUSHAB  P.ABM
                           69           02  FB 00245         CALLS   #2, CLI$PRESENT
                           56           50  D0 00248         MOVL    R0, STATUS
                00000000G  8F           56  D1 0024B         CMPL    STATUS, #CLI$_ABSENT              ; 0726
                                        4E  13 00252         BEQL    29$
                      08   A7           10  88 00254         BISB2   #16, SETFILE$JFLAGS               ; 0729
                           06           56  E9 00258         BLBC    STATUS, 26$                       ; 0730
                      08   A7           08  88 0025B         BISB2   #8, SETFILE$JFLAGS                ; 0732
                                        09  11 0025F         BRB     27$
                           5B           56  D1 00261 26$:    CMPL    STATUS, R11                       ; 0733
                                        04  12 00264         BNEQ    27$
                      08   A7           08  8A 00266         BICB2   #8, SETFILE$JFLAGS                ; 0735
                                    08  AE  9F 0026A 27$:    PUSHAB  DESC                              ; 0736
                                  017C  C8  9F 0026D         PUSHAB  P.ABO
```

```
                        6A          02 FB 00271          CALLS   #2, CLI$GET_VALUE
                        2B          50 E9 00274          BLBC    R0, 29$
                        10    08    AE B1 00277          CMPW    DESC, #16                                   ;0739
                              12    1B 0027B             BLEQU   28$
                          08    AE 9F 0027D              PUSHAB  DESC                                        ;0741
                              01    DD 00280             PUSHL   #1
                        00771112    8F DD 00282          PUSHL   #7803154
              00000000G 00          03 FB 00288          CALLS   #3, LIB$SIGNAL
                   09   A7      40 8F 88 0028F 28$:     BISB2   #64, SETFILE$JFLAGS+1                      ;0742
                 0178   C7    08    AE B0 00294          MOVW    DESC, AT_JNL_DESC                           ;0743
0150   C7          0C   BE    08    AE 28 0029A          MOVC3   DESC, @DESC+4, AT_JNL_NAME                  ;0744
                          08    AE 9F 002A2 29$:        PUSHAB  DESC                                        ;0751
                        0190    C8 9F 002A5              PUSHAB  P.ABQ
                        69          02 FB 002A9          CALLS   #2, CLI$PRESENT
                        56          50 D0 002AC          MOVL    R0, STATUS
              00000000G 8F          56 D1 002AF          CMPL    STATUS, #CLI$_ABSENT                        ;0752
                                    4F 13 002B6          BEQL    33$
                   08   A7      40 8F 88 002B8          BISB2   #64, SETFILE$JFLAGS                         ;0755
                        06          56 E9 002BD          BLBC    STATUS, 30$                                 ;0756
                   08   A7          20 88 002C0          BISB2   #32, SETFILE$JFLAGS                         ;0758
                                    09 11 002C4          BRB     31$
                        5B          56 D1 002C6 30$:     CMPL    STATUS, R11                                 ;0759
                                    04 12 002C9          BNEQ    31$
                   08   A7          20 8A 002CB          BICB2   #32, SETFILE$JFLAGS                         ;0761
                          08    AE 9F 002CF 31$:        PUSHAB  DESC                                        ;0762
                        01A4    C8 9F 002D2              PUSHAB  P.ABS
                        6A          02 FB 002D6          CALLS   #2, CLI$GET_VALUE
                        2B          50 E9 002D9          BLBC    R0, 33$
                        10    08    AE B1 002DC          CMPW    DESC, #16                                   ;0765
                              12    1B 002E0             BLEQU   32$
                          08    AE 9F 002E2              PUSHAB  DESC                                        ;0767
                              01    DD 002E5             PUSHL   #1
                        00771112    8F DD 002E7          PUSHL   #7803154
              00000000G 00          03 FB 002ED          CALLS   #3, LIB$SIGNAL
                   09   A7      80 8F 88 002F4 32$:     BISB2   #128, SETFILE$JFLAGS+1                     ;0768
                 0180   C7    08    AE B0 002F9          MOVW    DESC, BI_JNL_DESC                           ;0769
0160   C7          0C   BE    08    AE 28 002FF          MOVC3   DESC, @DESC+4, BI_JNL_NAME                  ;0770
                          08    AE 9F 00307 33$:        PUSHAB  DESC                                        ;0777
                        01B8    C8 9F 0030A              PUSHAB  P.ABU
                        69          02 FB 0030E          CALLS   #2, CLI$PRESENT
                        56          50 D0 00311          MOVL    R0, STATUS
              00000000G 8F          56 D1 00314          CMPL    STATUS, #CLI$_ABSENT                        ;0778
                                    18 13 0031B          BEQL    35$
                   09   A7          01 88 0031D          BISB2   #1, SETFILE$JFLAGS+1                        ;0781
                        07          56 E9 00321          BLBC    STATUS, 34$                                 ;0782
                   08   A7      80 8F 88 00324          BISB2   #128, SETFILE$JFLAGS                        ;0784
                                    0A 11 00329          BRB     35$
                        5B          56 D1 0032B 34$:     CMPL    STATUS, R11                                 ;0785
                                    05 12 0032E          BNEQ    35$
                   08   A7      80 8F 8A 00330          BICB2   #128, SETFILE$JFLAGS                        ;0787
                          08    AE 9F 00335 35$:        PUSHAB  DESC                                        ;0793
                        01D0    C8 9F 00338              PUSHAB  P.ABW
                        69          02 FB 0033C          CALLS   #2, CLI$PRESENT
                        56          50 D0 0033F          MOVL    R0, STATUS
              00000000G 8F          56 D1 00342          CMPL    STATUS, #CLI$_ABSENT                        ;0794
                                    16 13 00349          BEQL    37$
                   09   A7          10 88 0034B          BISB2   #16, SETFILE$JFLAGS+1                       ;0797
```

```
                              06          56  E9 0034F          BLBC    STATUS, 36$                          ; 0798
                       09     A7          08  88 00352          BISB2   #8, SETFILE$JFLAGS+1                 ; 0800
                                          09  11 00356          BRB     37$                                  ;
                              5B          56  D1 00358 36$:     CMPL    STATUS, R11                          ; 0801
                                          04  12 0035B          BNEQ    37$                                  ;
                       09     A7          08  8A 0035D          BICB2   #8, SETFILE$JFLAGS+1                 ; 0803
                                    08    AE  9F 00361 37$:     PUSHAB  DESC                                 ; 0809
                                  01E8    C8  9F 00364          PUSHAB  P.ABY                                ;
                              69          02  FB 00368          CALLS   #2, CLI$PRESENT                      ;
                              56          50  D0 0036B          MOVL    R0, STATUS                           ;
                00000000G     8F          56  D1 0036E          CMPL    STATUS, #CLI$_ABSENT                 ; 0810
                                          16  13 00375          BEQL    39$                                  ;
                       09     A7          04  88 00377          BISB2   #4, SETFILE$JFLAGS+1                 ; 0813
                              06          56  E9 0037B          BLBC    STATUS, 38$                          ; 0814
                       09     A7          02  88 0037E          BISB2   #2, SETFILE$JFLAGS+1                 ; 0816
                                          09  11 00382          BRB     39$                                  ;
                              5B          56  D1 00384 38$:     CMPL    STATUS, R11                          ; 0817
                                          04  12 00387          BNEQ    39$                                  ;
                       09     A7          02  8A 00389          BICB2   #2, SETFILE$JFLAGS+1                 ; 0819
                                  01F4    C8  9F 0038D 39$:     PUSHAB  P.ACA                                ; 0828
                              69          01  FB 00391          CALLS   #1, CLI$PRESENT                      ;
  01    A7           01       04          50  F0 00394          INSV    R0, #4, #1, SETFILE$FLAGS+1          ;
                                  0208    C8  9F 0039A          PUSHAB  P.ACC                                ; 0833
                              69          01  FB 0039E          CALLS   #1, CLI$PRESENT                      ;
                              04          50  E9 003A1          BLBC    R0, 40$                              ;
                       01     A7          20  88 003A4          BISB2   #32, SETFILE$FLAGS+1                 ; 0834
                                  021C    C8  9F 003A8 40$:     PUSHAB  P.ACE                                ; 0839
                              69          01  FB 003AC          CALLS   #1, CLI$PRESENT                      ;
                              5A          50  E9 003AF          BLBC    R0, 45$                              ;
                       01     A7    40    8F  88 003B2          BISB2   #64, SETFILE$FLAGS+1                 ; 0842
                                    08    AE  9F 003B7          PUSHAB  DESC                                 ; 0843
                                  0230    C8  9F 003BA          PUSHAB  P.ACG                                ;
                              6A          02  FB 003BE          CALLS   #2, CLI$GET_VALUE                    ;
                              1F          50  E8 003C1          BLBS    R0, 41$                              ;
                                          7E  7C 003C4          CLRQ    -(SP)                                ; 0852
                                    08    AE  9F 003C6          PUSHAB  IOSB                                 ;
                                  0238    C8  9F 003C9          PUSHAB  P.ACI                                ;
                                          7E  7C 003CD          CLRQ    -(SP)                                ;
                                          7E  D4 003CF          CLRL    -(SP)                                ;
                00000000G     00          07  FB 003D1          CALLS   #7, SYS$GETJPIW                      ;
                              56          50  D0 003D8          MOVL    R0, STATUS                           ;
                              29          56  E9 003DB          BLBC    STATUS, 44$                          ; 0853
                              56          6E  3C 003DE          MOVZWL  IOSB, STATUS                         ; 0854
                                          21  11 003E1          BRB     43$                                  ; 0855
          0248    C8   0C     BE    08    AE  29 003E3 41$:     CMPC3   DESC, @DESC+4, P.ACJ                 ; 0864
                                          07  12 003EB          BNEQ    42$                                  ;
                       01     A7    80    8F  88 003ED          BISB2   #128, SETFILE$FLAGS+1                ; 0866
                                          18  11 003F2          BRB     45$                                  ;
                                    28    A7  9F 003F4 42$:     PUSHAB  UIC_VALUE                            ; 0867
                                    0C    AE  9F 003F7          PUSHAB  DESC                                 ;
                00000000G     00          02  FB 003FA          CALLS   #2, PARSE_UIC                        ;
                              56          50  D0 00401          MOVL    R0, STATUS                           ;
                              05          56  E8 00404 43$:     BLBS    STATUS, 45$                          ;
                                          56  DD 00407 44$:     PUSHL   STATUS                               ; 0870
                                        016B  31 00409          BRW     60$                                  ;
                                  025C    C8  9F 0040C 45$:     PUSHAB  P.ACK                                ; 0878
                              69          01  FB 00410          CALLS   #1, CLI$PRESENT                      ;
```

```
                 03                50  E8 00413          BLBS    R0, 46$                          :
                             00CB  31 00416          BRW     52$                              :
           02    A7                01  88 00419 46$:     BISB2   #1, SETFILE$FLAGS+2              : 0884
                 6E  020E0000      8F  D0 0041D          MOVL    #34471936, PROT_DESC             : 0889
                           04      AE  D4 00424          CLRL    PROT_DESC+4                      :
                         0278      C8  9F 00427          PUSHAB  P.ACM                            : 0891
                 69                01  FB 0042B          CALLS   #1, CLI$PRESENT                  :
                 1D                50  E9 0042E          BLBC    R0, 47$                          :
           12    A7                0F  88 00431          BISB2   #15, SETPRO_MASK                 : 0894
                                   5E  DD 00435          PUSHL   SP                               : 0895
                         0294      C8  9F 00437          PUSHAB  P.ACO                            :
                 6A                02  FB 0043B          CALLS   #2, CLI$GET_VALUE                :
                 0D                50  E9 0043E          BLBC    R0, 47$                          :
                                   5E  DD 00441          PUSHL   SP                               : 0896
      00000000V  EF                01  FB 00443          CALLS   #1, PARSE_CLASS                  :
           10    A7                50  B0 0044A          MOVW    R0, SETPRO_PROT                  :
                         02AC      C8  9F 0044E 47$:     PUSHAB  P.ACQ                            : 0898
                 69                01  FB 00452          CALLS   #1, CLI$PRESENT                  :
                 21                50  E9 00455          BLBC    R0, 48$                          :
           12    A7          F0    8F  88 00458          BISB2   #240, SETPRO_MASK                : 0901
                                   5E  DD 0045D          PUSHL   SP                               : 0902
                         02C4      C8  9F 0045F          PUSHAB  P.ACS                            :
                 6A                02  FB 00463          CALLS   #2, CLI$GET_VALUE                :
                 10                50  E9 00466          BLBC    R0, 48$                          :
                                   5E  DD 00469          PUSHL   SP                               : 0903
      00000000V  EF                01  FB 0046B          CALLS   #1, PARSE_CLASS                  :
                 50                10  C4 00472          MULL2   #16, R0                          :
           10    A7                50  A8 00475          BISW2   R0, SETPRO_PROT                  :
                         02DC      C8  9F 00479 48$:     PUSHAB  P.ACU                            : 0905
                 69                01  FB 0047D          CALLS   #1, CLI$PRESENT                  :
                 21                50  E9 00480          BLBC    R0, 49$                          :
           13    A7                0F  88 00483          BISB2   #15, SETPRO_MASK+1               : 0908
                                   5E  DD 00487          PUSHL   SP                               : 0909
                         02F4      C8  9F 00489          PUSHAB  P.ACW                            :
                 6A                02  FB 0048D          CALLS   #2, CLI$GET_VALUE                :
                 11                50  E9 00490          BLBC    R0, 49$                          :
                                   5E  DD 00493          PUSHL   SP                               : 0910
      00000000V  EF                01  FB 00495          CALLS   #1, PARSE_CLASS                  :
 50              50                08  78 0049C          ASHL    #8, R0, R0                       :
           10    A7                50  A8 004A0          BISW2   R0, SETPRO_PROT                  :
                         030C      C8  9F 004A4 49$:     PUSHAB  P.ACY                            : 0912
                 69                01  FB 004A8          CALLS   #1, CLI$PRESENT                  :
                 22                50  E9 004AB          BLBC    R0, 50$                          :
           13    A7          F0    8F  88 004AE          BISB2   #240, SETPRO_MASK+1              : 0915
                                   5E  DD 004B3          PUSHL   SP                               : 0916
                         0324      C8  9F 004B5          PUSHAB  P.ADA                            :
                 6A                02  FB 004B9          CALLS   #2, CLI$GET_VALUE                :
                 11                50  E9 004BC          BLBC    R0, 50$                          :
                                   5E  DD 004BF          PUSHL   SP                               : 0917
      00000000V  EF                01  FB 004C1          CALLS   #1, PARSE_CLASS                  :
 50              50                0C  78 004C8          ASHL    #12, R0, R0                      :
           10    A7                50  A8 004CC          BISW2   R0, SETPRO_PROT                  :
                 50          12    A7  3C 004D0 50$:     MOVZWL  SETPRO_MASK, R0                  : 0925
                                   05  13 004D4          BEQL    51$                              :
           10    A7          10    A7  B2 004D6          MCOMW   SETPRO_PROT, SETPRO_PROT         : 0926
           16    A7                50  B0 004DB 51$:     MOVW    R0, GLOBAL_MASK                  : 0933
           14    A7          10    A7  B0 004DF          MOVW    SETPRO_PROT, GLOBAL_PROT         : 0934
```

```
                        0334  C8  9F 004E4 52$:    PUSHAB  P.ADC                           ; 0939
                          69  01  FB 004E8         CALLS   #1, CLI$PRESENT                 ;
                          04  50  E9 004EB         BLBC    R0, 53$                         ;
                    02    A7  20  88 004EE         BISB2   #32, SETFILE$FLAGS+2            ; 0940
                        0344  C8  9F 004F2 53$:    PUSHAB  P.ADE                           ; 0945
                          69  01  FB 004F6         CALLS   #1, CLI$PRESENT                 ;
                          04  50  E9 004F9         BLBC    R0, 54$                         ;
                    02    A7  02  88 004FC         BISB2   #2, SETFILE$FLAGS+2             ; 0946
                        0354  C8  9F 00500 54$:    PUSHAB  P.ADG                           ; 0951
                          69  01  FB 00504         CALLS   #1, CLI$PRESENT                 ;
                          04  50  E9 00507         BLBC    R0, 55$                         ;
                    02    A7  04  88 0050A         BISB2   #4, SETFILE$FLAGS+2             ; 0952
                        036C  C8  9F 0050E 55$:    PUSHAB  P.ADI                           ; 0957
                          69  01  FB 00512         CALLS   #1, CLI$PRESENT                 ;
                          68  50  E9 00515         BLBC    R0, 61$                         ;
                    02    A7  08  88 00518         BISB2   #8, SETFILE$FLAGS+2             ; 0960
                    34    A7  7FFF 8F 3C 0051C     MOVZWL  #32767, VRSN_VALUE              ; 0961
                              08  AE 9F 00522       PUSHAB  DESC                            ; 0962
                        0384  C8  9F 00525         PUSHAB  P.ADK                           ;
                          6A  02  FB 00529         CALLS   #2, CLI$GET_VALUE               ;
                          51  50  E9 0052C         BLBC    R0, 61$                         ;
                                34  A7 9F 0052F    PUSHAB  VRSN_VALUE                      ; 0965
                                10  AE DD 00532    PUSHL   DESC+4                          ; 0966
                              7E    10  AE 3C 00535 MOVZWL  DESC, -(SP)                    ; 0965
                00000000G     00  03  FB 00539     CALLS   #3, LIB$CVT_DTB                 ;
                              14  50  E8 00540     BLBS    R0, 57$                         ;
                                08  AE 9F 00543 56$:  PUSHAB  DESC                        ; 0970
                                01  DD 00546       PUSHL   #1                              ;
                          007710FA  8F DD 00548     PUSHL   #7803130                        ;
                00000000G     00  03  FB 0054E     CALLS   #3, LIB$SIGNAL                  ;
                                2D  11 00555       BRB     62$                             ; 0971
                                34  A7 D5 00557 57$:  TSTL  VRSN_VALUE                      ; 0974
                                06  12 0055A       BNEQ    58$                             ;
                            34  A7      7FFF 8F 3C 0055C MOVZWL  #32767, VRSN_VALUE        ; 0976
                              50        34  A7 D0 00562 58$:  MOVL  VRSN_VALUE, R0          ; 0978
                                09  19 00566       BLSS    59$                             ;
                00007FFF      8F  50  D1 00568     CMPL    R0, #32767                      ; 0979
                                0F  15 0056F       BLEQ    61$                             ;
                          007711EA  8F DD 00571 59$:  PUSHL  #7803370                        ; 0981
                00000000G     00  01  FB 00577 60$:  CALLS  #1, LIB$SIGNAL                  ;
                                04  11 0057E       BRB     62$                             ;
                              50    01  D0 00580 61$:  MOVL  #1, R0                        ; 0985
                                    04 00583       RET                                     ;
                                50  D4 00584 62$:  CLRL  R0                                ; 0986
                                    04 00586       RET                                     ;

; Routine Size:  1415 bytes,    Routine Base:  $CODE$ + 00E0
```

```
;   994          0987  1 ROUTINE parse_null_string (fab) : NOVALUE =
;   995          0988  1 !++
;   996          0989  1 !
;   997          0990  1 ! This routine parses the null string on the specified FAB to
;   998          0991  1 ! force RMS to clear all internal saved context.
;   999          0992  1 !
;  1000          0993  1 !--
;  1001          0994  2 BEGIN
;  1002          0995  2 MAP
;  1003          0996  2     fab : REF $BBLOCK;
;  1004          0997  2
;  1005          0998  2 LOCAL
;  1006          0999  2     nam : REF $BBLOCK;
;  1007          1000  2
;  1008          1001  2 nam = .fab[fab$l_nam];
;  1009          1002  2 nam[nam$v_svctx] = 0;
;  1010          1003  2 nam[nam$v_synchk] = 1;
;  1011          1004  2 nam[nam$b_esl] = 0;
;  1012          1005  2 nam[nam$b_ess] = 0;
;  1013          1006  2 nam[nam$b_rsl] = 0;
;  1014          1007  2 nam[nam$b_rss] = 0;
;  1015          1008  2 nam[nam$l_esa] = 0;
;  1016          1009  2 nam[nam$l_rsa] = 0;
;  1017          1010  2 nam[nam$l_rlf] = 0;
;  1018          1011  2 fab[fab$b_fns] = 0;
;  1019          1012  2 fab[fab$b_dns] = 0;
;  1020          1013  2 $parse(fab=.fab);
;  1021          1014  2 RETURN;
;  1022          1015  1 END;


                                                .EXTRN  SYS$PARSE

                                0000 00000 PARSE_NULL_STRING:
                                                .WORD   Save nothing                    ; 0987
                          51    04   AC  D0 00002          MOVL    FAB, R1                 ; 1001
                          50    28   A1  D0 00006          MOVL    40(R1), NAM             ;
                33        A0    80   8F  8A 0000A          BICB2   #128, 51(NAM)           ; 1002
                08        A0          08  88 0000F          BISB2   #8, 8(NAM)              ; 1003
                                      0A  A0 B4 00013       CLRW    10(NAM)                 ; 1005
                                      02  A0 B4 00016       CLRW    2(NAM)                  ; 1007
                                      04  A0 D4 00019       CLRL    4(NAM)                  ; 1009
                                      0C  A0 7C 0001C       CLRQ    12(NAM)                 ; 1008
                                      34  A1 B4 0001F       CLRW    52(R1)                  ; 1011
                                          51 DD 00022       PUSHL   R1                      ; 1013
                    00000000G     00        01 FB 00024     CALLS   #1, SYS$PARSE           ;
                                            04 0002B        RET                             ; 1015

; Routine Size:  44 bytes,    Routine Base:  $CODE$ + 0667
```

```
;  1024          1016  1 ROUTINE set_attributes (fab) =
;  1025          1017  1 !++
;  1026          1018  1 !
;  1027          1019  1 ! This is the routine that actually accesses the file, and sets the
;  1028          1020  1 ! specified attributes.  If an error occurs while attempting to set
;  1029          1021  1 ! the attributes, a message telling the user is issued, and any other
;  1030          1022  1 ! files are processed.
;  1031          1023  1 !
;  1032          1024  1 !--
;  1033          1025  2 BEGIN
;  1034          1026  2
;  1035          1027  2 MAP
;  1036          1028  2     fab : REF $BBLOCK;                     ! Define the fab
;  1037          1029  2
;  1038          1030  2
;  1039          1031  2
;  1040          1032  2 LOCAL
;  1041          1033  2     atr : BLOCKVECTOR[13,8,BYTE],          ! Attribute control block
;  1042          1034  2     ptr,                                   ! Pointer to attribute block
;  1043          1035  2     status,                                ! Status return
;  1044          1036  2     channel : WORD,                        ! Channel number
;  1045          1037  2     desc : $BBLOCK[dsc$c_s_bln],                   ! General descriptor
;  1046          1038  2     fib : $BBLOCK[fib$c_length],                   ! A FIB for the QIO
;  1047          1039  2     header : $BBLOCK[512],                 ! The file header
;  1048          1040  2     item_list : $ITMLST_DECL (ITEMS=1),    ! Item list for GETDVI volume label
;  1049          1041  2     ai_jnl_ace : $BBLOCK[4+cjf$c_maxjnlnaml],      ! ACE to contain AI journal name
;  1050          1042  2     at_jnl_ace : $BBLOCK[4+cjf$c_maxjnlnaml],      ! ACE to contain AT journal name
;  1051          1043  2     bi_jnl_ace : $BBLOCK[4+cjf$c_maxjnlnaml],      ! ACE to contain BI journal name
;  1052          1044  2     label_buffer : VECTOR[12,BYTE],                ! Buffer for volume label
;  1053          1045  2     iosb : VECTOR[4,WORD];                 ! I/O status block
;  1054          1046  2
;  1055          1047  2 BIND
;  1056          1048  2     recattr = header[fh2$w_recattr] : $BBLOCK[atr$s_recattr],
;  1057          1049  2
;  1058          1050  2     nam = .fab[fab$l_nam] : $BBLOCK;       ! Define the name block
;  1059          1051  2
;  1060          1052  2 OWN
;  1061          1053  2     rmsjnlid_ace : $BBLOCK[id_ace$s_size],         ! ACE to contain RMS jnl ident
;  1062          1054  2     old_did_num : WORD,                            ! Old directory id
;  1063          1055  2     old_did_seq : WORD,
;  1064          1056  2     old_did_rvn : WORD;
;  1065          1057  2
;  1066          1058  2 !
;  1067          1059  2 ! If no more processing is to be performed, then simply return.  This handles
;  1068          1060  2 ! the case of wildcards which do not return for each file to the routine
;  1069          1061  2 ! that called lib$file_scan.
;  1070          1062  2 !
;  1071          1063  2 IF .setfile$flags[qual_quit]
;  1072          1064  2 THEN RETURN true;
;  1073          1065  2
;  1074          1066  2 !
;  1075          1067  2 ! See if the common specified common qualifiers match this one.
;  1076          1068  2 !
;  1077          1069  2 conf_desc[dsc$a_pointer] = .nam[nam$l_rsa];       ! Get the resultant name
;  1078          1070  2 conf_desc[dsc$w_length] = .nam[nam$b_rsl];       ! of this file
;  1079          1071  2
;  1080          1072  2 status = lib$qual_file_match(context,           ! Call the common qualifier routine
```

```
;  1081     1073  2                                   0,
;  1082     1074  2                                   conf_desc,
;  1083     1075  2                                   0,
;  1084     1076  2                                   0,
;  1085     1077  2                                   0);
;  1086     1078  2          IF NOT .status                          ! If the status is false, check
;  1087     1079  2          THEN                                    ! if it is the "correct" false
;  1088     1080  2              BEGIN                               ! errors that says
;  1089     1081  2              IF .status NEQ lib$_filfaimat       ! "file didn't match qualifiers"
;  1090     1082  2              THEN SIGNAL(set$_openin,            ! Else signal it
;  1091     1083  2                          1,
;  1092     1084  2                          conf_desc,
;  1093     1085  2                          .status);
;  1094     1086  2              RETURN true;                        ! and DON'T process this file
;  1095     1087  2              END;
;  1096     1088  2
;  1097     1089  2
;  1098     1090  2          !
;  1099     1091  2          ! Assign a channel to the file's device
;  1100     1092  2          !
;  1101     1093  2          desc[dsc$w_length] = .nam[nam$b_dev];   ! Set up the descriptor
;  1102     1094  2          desc[dsc$a_pointer] = .nam[nam$t_dev];  ! to point to the device name
;  1103   P 1095  2          IF NOT (status = $ASSIGN(
;  1104   P 1096  2                                  DEVNAM = desc,
;  1105     1097  2                                  CHAN = channel))
;  1106     1098  2          THEN
;  1107     1099  2              BEGIN
;  1108     1100  2              file_error( set$_openin,
;  1109     1101  2                          .status, .fab);         ! Tell user why the assign failed
;  1110     1102  2              RETURN true;                        ! And continue with other files
;  1111     1103  2              END;
;  1112     1104  2
;  1113     1105  2          !
;  1114     1106  2          ! Access the file, reading the file's header
;  1115     1107  2          !
;  1116     1108  2          desc[dsc$w_length] = fib$c_length;      ! Re-use descriptor to point to FIB
;  1117     1109  2          desc[dsc$a_pointer] = fib;
;  1118     1110  2
;  1119     1111  2          CH$FILL(0,fib$c_length,fib);            ! Zero out the FIB
;  1120     1112  2
;  1121     1113  2          fib[fib$l_acctl] = fib$m_write OR       ! Set up the FIB
;  1122     1114  2                             fib$m_noread OR
;  1123     1115  2                             fib$m_nowrite;
;  1124     1116  2
;  1125     1117  2          fib[fib$w_fid_num] = .nam[nam$w_fid_num];       ! Put in the file id
;  1126     1118  2          fib[fib$w_fid_seq] = .nam[nam$w_fid_seq];
;  1127     1119  2          fib[fib$w_fid_rvn] = .nam[nam$w_fid_rvn];
;  1128     1120  2
;  1129     1121  2
;  1130     1122  2          !
;  1131     1123  2          ! Unless some option was specified that requires the file header, don't
;  1132     1124  2          ! bother to get it.
;  1133     1125  2          !
;  1134     1126  3          IF (.setfile$flags[qual_backup] OR
;  1135     1127  3              .setfile$flags[qual_nobackup] OR
;  1136     1128  3              .setfile$flags[qual_data] OR
;  1137     1129  3              .setfile$flags[qual_eof] OR
```

```
;  1138          1130  3      .setfile$flags[qual_erase] OR
;  1139          1131  3      .setfile$flags[qual_noerase] OR
;  1140          1132  3      .setfile$flags[qual_expi] OR
;  1141          1133  3      .setfile$flags[qual_exte] OR
;  1142          1134  3      .setfile$flags[qual_gbuf] OR
;  1143          1135  3      .setfile$flags[qual_journal] OR
;  1144          1136  3      .setfile$flags[qual_nodi] OR
;  1145          1137  3      .setfile$flags[qual_owner] OR
;  1146          1138  3      .setfile$flags[qual_trunc] OR
;  1147          1139  3      .setfile$flags[qual_vrsn])
;  1148          1140  2  AND
;  1149          1141  3      BEGIN
;  1150          1142  3      IF .setfile$flags[qual_quit_mod]
;  1151          1143  3      OR NOT .setfile$flags[qual_confirm]
;  1152          1144  3      THEN true
;  1153          1145  3      ELSE
;  1154          1146  4          BEGIN
;  1155          1147  4          status = lib$confirm_act(%ASCID 'Modify file !AS? [N] : ',
;  1156          1148  4                                   %REF(conf_desc));
;  1157          1149  4          IF NOT .status
;  1158          1150  4          THEN
;  1159          1151  5              BEGIN
;  1160          1152  5              IF .status EQL lib$_quipro
;  1161          1153  6              THEN (setfile$flags[qual_quit] = 1; RETURN true)
;  1162          1154  5              ELSE IF .status EQL lib$_quiconact
;  1163          1155  6              THEN (setfile$flags[qual_quit_mod] = 1; status = 1)
;  1164          1156  5              ELSE IF .status NEQ lib$_negans
;  1165          1157  5              THEN SIGNAL(set$_writeerr, 1, conf_desc, .status);
;  1166          1158  4              END;
;  1167          1159  4          .status
;  1168          1160  4          END
;  1169          1161  3      END
;  1170          1162  2  THEN
;  1171          1163  3  BEGIN
;  1172          1164  3
;  1173          1165  3  atr[0,atr$w_type] = atr$c_header;                ! Get the file header
;  1174          1166  3  atr[0,atr$w_size] = atr$s_header;
;  1175          1167  3  atr[0,atr$l_addr] = header;
;  1176          1168  3  atr[1,atr$w_type] = atr$c_fndacetyp;            ! Look for an rmsjnlid ACE
;  1177          1169  3  atr[1,atr$w_size] = id_ace$s_size;
;  1178          1170  3  atr[1,atr$l_addr] = rmsjnlid_ace;
;  1179          1171  3  atr[2,0,0,32,0] = 0;
;  1180          1172  3
;  1181          1173  3  rmsjnlid_ace[ace$b_size] = id_ace$s_size;
;  1182          1174  3  rmsjnlid_ace[ace$b_type] = ace$c_jnlid;
;  1183          1175  3  rmsjnlid_ace[ace$w_flags] = ace$m_hidden OR ace$m_protected;
;  1184          1176  3
;  1185        P 1177  3  status = $QIOW( CHAN = .channel,
;  1186        P 1178  3                  FUNC = IO$_ACCESS OR IO$M_ACCESS,
;  1187        P 1179  3                  IOSB = iosb,
;  1188        P 1180  3                  P1 = desc,
;  1189          1181  3                  P5 = atr);
;  1190          1182  3  IF .status THEN status = .iosb[0];
;  1191          1183  3  IF NOT .status
;  1192          1184  3  THEN file_error(set$_readerr,,status,,fab)
;  1193          1185  3  ELSE
;  1194          1186  4      BEGIN
```

```
;  1195          1187  4 !
;  1196          1188  4 ! Check to see whether this is an ODS1 or an ODS2 file.  If ODS1,
;  1197          1189  4 ! copy the record attributes into the ODS2 location.
;  1198          1190  4 !
;  1199          1191  4     IF .header[fh2$b_struclev] EQL 1
;  1200          1192  4     THEN CH$MOVE(fat$c_length, header[fh1$w_recattr], header[fh2$w_recattr]);
;  1201          1193  4
;  1202          1194  4 !
;  1203          1195  4 ! See if an rmsjnlid ACE already exists.  Don't create a new one later, since
;  1204          1196  4 ! one is good enough.
;  1205          1197  4 !
;  1206          1198  4     IF .rmsjnlid_ace[ace$b_type] NEQ 0
;  1207          1199  4     THEN
;  1208          1200  4         setfile$mflags[misc_already_rmsjnlid] = 1;
;  1209          1201  4
;  1210          1202  4 !
;  1211          1203  4 ! Initialize the pointer to the attribute control block.  The block
;  1212          1204  4 ! will be built as we go, and the pointer shows where the next attribute
;  1213          1205  4 ! should go in the block.
;  1214          1206  4 !
;  1215          1207  4     ptr = 0;
;  1216          1208  4
;  1217          1209  4 !
;  1218          1210  4 ! Change the file characteristics
;  1219          1211  4 !
;  1220          1212  4     status = 0;                  ! Show that nothing has changed
;  1221          1213  4
;  1222          1214  4     IF .setfile$flags[qual_backup]              ! /BACKUP
;  1223          1215  4     THEN
;  1224          1216  5         BEGIN
;  1225          1217  5         header[fh2$v_nobackup] = 0;
;  1226          1218  5         status = 1;
;  1227          1219  4         END;
;  1228          1220  4     IF .setfile$flags[qual_nobackup]            ! /NOBACKUP
;  1229          1221  4     THEN
;  1230          1222  5         BEGIN
;  1231          1223  5         header[fh2$v_nobackup] = 1;
;  1232          1224  5         status = 1;
;  1233          1225  4         END;
;  1234          1226  4
;  1235          1227  4     IF .setfile$flags[qual_erase]               ! /ERASE
;  1236          1228  4     THEN
;  1237          1229  5         BEGIN
;  1238          1230  5         IF .header[fh2$b_struclev] EQL 1        ! If not ODS2
;  1239          1231  5         THEN SIGNAL(set$_notods2,               ! tell the user
;  1240          1232  5                     1,
;  1241          1233  5                     $DESCRIPTOR('/ERASE'))
;  1242          1234  5         ELSE
;  1243          1235  6             BEGIN
;  1244          1236  6             header[fh2$v_erase] = 1;
;  1245          1237  6             status = 1;
;  1246          1238  5             END;
;  1247          1239  4         END;
;  1248          1240  4     IF .setfile$flags[qual_noerase]             ! /NOERASE
;  1249          1241  4     THEN
;  1250          1242  5         BEGIN
;  1251          1243  5         IF .header[fh2$b_struclev] EQL 1        ! If not ODS2
```

```
;  1252     1244  5          THEN SIGNAL(set$_notods2,                    ! tell the user
;  1253     1245  5                      1,
;  1254     1246  5                      $DESCRIPTOR('/NOERASE'))
;  1255     1247  5          ELSE
;  1256     1248  6              BEGIN
;  1257     1249  6              header[fh2$v_erase] = 0;
;  1258     1250  6              status = 1;
;  1259     1251  5              END;
;  1260     1252  4          END;
;  1261     1253  4
;  1262     1254  4      IF .setfile$flags[qual_data]                     ! /DATA_CHECK
;  1263     1255  4      THEN
;  1264     1256  5          BEGIN
;  1265     1257  5          IF .setfile$dflags[data_read] THEN header[fh2$v_readcheck] = 1;
;  1266     1258  5          IF .setfile$dflags[data_noread] THEN header[fh2$v_readcheck] = 0;
;  1267     1259  5          IF .setfile$dflags[data_write] THEN header[fh2$v_writcheck]= 1;
;  1268     1260  5          IF .setfile$dflags[data_nowrite] THEN header[fh2$v_writcheck] =0;
;  1269     1261  5          status = 1;
;  1270     1262  4          END;
;  1271     1263  4
;  1272     1264  4      IF .setfile$flags[qual_nodi]                     ! /NODIRECTORY
;  1273     1265  4      THEN
;  1274     1266  5          BEGIN
;  1275     1267  5          IF .header[fh2$b_strucLev] EQL 1             ! If not ODS2
;  1276     1268  5          THEN SIGNAL (set$_notods2,                   ! tell the user
;  1277     1269  5                      1,
;  1278     1270  5                      $DESCRIPTOR('/NODIRECTORY'))
;  1279     1271  5          ELSE
;  1280     1272  6              BEGIN
;  1281     1273  6              header[fh2$v_directory] = 0;
;  1282     1274  6              status = 1;
;  1283     1275  5              END;
;  1284     1276  4          END;
;  1285     1277  4
;  1286     1278  4      !
;  1287     1279  4      ! If something in the file characteristics was changed, show it.
;  1288     1280  4      !
;  1289     1281  4          IF .status
;  1290     1282  4          THEN
;  1291     1283  5              BEGIN
;  1292     1284  5              atr[.ptr,atr$w_type] = atr$c_uchar;
;  1293     1285  5              atr[.ptr,atr$w_size] = atr$s_uchar;
;  1294     1286  5              atr[.ptr,atr$l_addr] = header[fh2$l_filechar];
;  1295     1287  5              ptr = .ptr + 1;
;  1296     1288  5              status = 0;                              ! Reset the change indicator
;  1297     1289  4              END;
;  1298     1290  4
;  1299     1291  4      !
;  1300     1292  4      ! Modify the record attributes
;  1301     1293  4      !
;  1302     1294  4
;  1303     1295  4          IF .setfile$flags[qual_exte]                 ! /EXTENSION
;  1304     1296  4          THEN
;  1305     1297  5              BEGIN
;  1306     1298  5              recattr[fat$w_defext] = .exte_value;
;  1307     1299  5              status = 1;
;  1308     1300  4              END;
```

```
;  1309          1301  4 !
;  1310          1302  4 ! IF /END_OF_FILE was specified, set the eof_block equal to the
;  1311          1303  4 ! highest block allocated, and the first free byte in that block
;  1312          1304  4 ! to 512, indicating that the entire allocated space is used.
;  1313          1305  4 !
;  1314          1306  4     IF .setfile$flags[qual_eof]
;  1315          1307  4     THEN
;  1316          1308  5         BEGIN
;  1317          1309  5         IF  .nam[nam$w_fid_num] EQL 1                   ! If INDEXF.SYS
;  1318          1310  5         AND .nam[nam$w_fid_seq] EQL 1
;  1319          1311  5         AND .nam[nam$b_fid_nmx] EQL 0
;  1320          1312  5         THEN SIGNAL (set$_writeerr,                     ! Signal an error
;  1321          1313  5                      1,                                 ! modifying it
;  1322          1314  5                      conf_desc,                         ! because of an
;  1323          1315  5                      ss$_acconflict)
;  1324          1316  5         ELSE
;  1325          1317  6             BEGIN
;  1326          1318  6             recattr[fat$l_efblk] = .recattr[fat$l_hiblk];
;  1327          1319  6             recattr[fat$w_ffbyte] = 512;
;  1328          1320  6             status = 1;
;  1329          1321  5             END;
;  1330          1322  4         END;
;  1331          1323  4 !
;  1332          1324  4 ! If /GLOBAL_BUFFERS was specified, set the global buffer count to
;  1333          1325  4 ! the value specified.
;  1334          1326  4 !
;  1335          1327  4     IF .setfile$flags[qual_gbuf]
;  1336          1328  4     THEN
;  1337          1329  5         BEGIN
;  1338          1330  5         recattr[fat$w_gbc] = .gbuf_value;
;  1339          1331  5         status = 1;
;  1340          1332  4         END;
;  1341          1333  4 !
;  1342          1334  4 ! If something in the user attributes was changed, show it.
;  1343          1335  4 !
;  1344          1336  4     IF .status
;  1345          1337  4     THEN
;  1346          1338  5         BEGIN
;  1347          1339  5         atr[.ptr,atr$w_type] = atr$c_recattr;
;  1348          1340  5         atr[.ptr,atr$w_size] = atr$s_recattr;
;  1349          1341  5         atr[.ptr,atr$l_addr] = header[fh2$w_recattr];
;  1350          1342  5         ptr = .ptr + 1;
;  1351          1343  4         END;
;  1352          1344  4
;  1353          1345  4 !
;  1354          1346  4 ! Expiration date
;  1355          1347  4 !
;  1356          1348  4     IF .setfile$flags[qual_expi]
;  1357          1349  4     THEN
;  1358          1350  5         BEGIN
;  1359          1351  5         CH$MOVE(8,exp_value,header[fi2$q_expdate]);
;  1360          1352  5         atr[.ptr,atr$w_type] = atr$c_expdate;
;  1361          1353  5         atr[.ptr,atr$w_size] = atr$s_expdate;
;  1362          1354  5         atr[.ptr,atr$l_addr] = header[fi2$q_expdate];
;  1363          1355  5         ptr = .ptr + 1;
;  1364          1356  4         END;
;  1365          1357  4
```

```
;  1366          1358  4 !
;  1367          1359  4 ! Owner UIC
;  1368          1360  4 !
;  1369          1361  4     IF .setfile$flags[qual_owner]
;  1370          1362  4     THEN
;  1371          1363  5         BEGIN
;  1372          1364  5 !
;  1373          1365  5 ! If the qualifier OWNER=PARENT was specified, then the UIC of the owner
;  1374          1366  5 ! directory must be found.  Rather than accessing the directory every time, a
;  1375          1367  5 ! test is made to determine if the directory's UIC has already been found.  If
;  1376          1368  5 ! so, then the current value of UIC_VALUE is used.  Otherwise, a new value is
;  1377          1369  5 ! found.
;  1378          1370  5 !
;  1379          1371  5         IF .setfile$flags[qual_parent]
;  1380          1372  5         THEN
;  1381          1373  6             BEGIN
;  1382          1374  7             IF NOT ((.nam[nam$w_did_num] EQL .old_did_num) AND
;  1383          1375  7                     (.nam[nam$w_did_seq] EQL .old_did_seq) AND
;  1384          1376  7                     (.nam[nam$w_did_rvn] EQL .old_did_rvn))
;  1385          1377  6             THEN
;  1386          1378  7                 BEGIN
;  1387          1379  7                 LOCAL
;  1388          1380  7                     temp_atr : BLOCKVECTOR[2,8,BYTE],
;  1389          1381  7                     temp_desc : $BBLOCK[dsc$c_s_bln],
;  1390          1382  7                     temp_fib : $BBLOCK[fib$c_extdata],
;  1391          1383  7                     temp_chan;
;  1392          1384  7
;  1393          1385  7                 temp_desc[dsc$w_length] = .nam[nam$b_dev];
;  1394          1386  7                 temp_desc[dsc$a_pointer] = .nam[nam$[_dev];
;  1395        P 1387  8                 IF NOT (status = $ASSIGN(DEVNAM = temp_desc,
;  1396          1388  8                                          CHAN =temp_chan))
;  1397          1389  7                 THEN
;  1398          1390  8                     BEGIN
;  1399          1391  8                     $DASSGN(CHAN = .channel);
;  1400          1392  8                     SIGNAL(set$_opendir, 1, conf_desc, .status);
;  1401          1393  8                     RETURN true;
;  1402          1394  7                     END;
;  1403          1395  7
;  1404          1396  7                 CH$FILL(0, fib$c_extdata, temp_fib);
;  1405          1397  7
;  1406          1398  7                 temp_fib[fib$l_acctl] = fib$m_noread OR fib$m_nowrite;
;  1407          1399  7                 temp_fib[fib$w_fid_num] = .nam[nam$w_did_num];
;  1408          1400  7                 temp_fib[fib$w_fid_seq] = .nam[nam$w_did_seq];
;  1409          1401  7                 temp_fib[fib$w_fid_rvn] = .nam[nam$w_did_rvn];
;  1410          1402  7
;  1411          1403  7                 temp_atr[0,atr$w_type] = atr$c_uic;
;  1412          1404  7                 temp_atr[0,atr$w_size] = atr$s_uic;
;  1413          1405  7                 temp_atr[0,atr$l_addr] = uic_value;
;  1414          1406  7                 temp_atr[1,0,0,32,0] = 0;
;  1415          1407  7
;  1416          1408  7                 temp_desc[dsc$w_length] = fib$c_extdata;
;  1417          1409  7                 temp_desc[dsc$a_pointer] = temp_fib;
;  1418          1410  7
;  1419        P 1411  7                 status = $QIOW( CHAN = .temp_chan,
;  1420        P 1412  7                                 FUNC = IO$_ACCESS,
;  1421        P 1413  7                                 IOSB = iosb,
;  1422        P 1414  7                                 P1 = temp_desc,
```

```
;  1423     1415  7                                  p5 = temp_atr);
;  1424     1416  7                          IF .status THEN status = .iosb[0];
;  1425     1417  7                          IF NOT .status
;  1426     1418  7                          THEN SIGNAL_STOP(set$_opendir, 1, conf_desc, .status);
;  1427     1419  7                          $DASSGN (CHAN = .temp_chan);
;  1428     1420  6                          END;
;  1429     1421  5                      END;
;  1430     1422  5
;  1431     1423  5                  header[fh2$l_fileowner] = .uic_value;
;  1432     1424  5                  atr[.ptr,atr$w_type] = atr$c_uic;
;  1433     1425  5                  atr[.ptr,atr$w_size] = atr$s_uic;
;  1434     1426  5                  atr[.ptr,atr$l_addr] = header[fh2$l_fileowner];
;  1435     1427  5                  ptr = .ptr + 1;
;  1436     1428  4                  END;
;  1437     1429  4
;  1438     1430  4  !
;  1439     1431  4  ! If /TRUNCATE was specified, find the block containing the EOF.  If
;  1440     1432  4  ! the EOF occurred somewhere in that block, then truncate to the next
;  1441     1433  4  ! block.
;  1442     1434  4  !
;  1443     1435  4      IF .setfile$flags[qual_trunc] THEN
;  1444     1436  5          BEGIN
;  1445     1437  5          IF .recattr[fat$v_fileorg] EQL fat$c_indexed
;  1446     1438  5          THEN
;  1447     1439  5              SIGNAL(set$_writeerr, 1, conf_desc, set$_notrunc)
;  1448     1440  5          ELSE
;  1449     1441  6              BEGIN
;  1450     1442  6              fib[fib$v_trunc] = 1;
;  1451     1443  6              fib[fib$l_exvbn] = .recattr[fat$w_efblkh]^16
;  1452     1444  6                                 + .recattr[fat$w_efblkl];
;  1453     1445  6              IF .recattr[fat$w_ffbyte] GTR 0
;  1454     1446  6              THEN fib[fib$l_exvbn] = .fib[fib$l_exvbn] + 1;
;  1455     1447  5              END;
;  1456     1448  4          END;
;  1457     1449  4  !
;  1458     1450  4  ! Set the version limit for a particular file
;  1459     1451  4  !
;  1460     1452  4      IF .setfile$flags[qual_vrsn]
;  1461     1453  4      THEN
;  1462     1454  5          BEGIN
;  1463     1455  5          fib[fib$w_did_num] = .nam[nam$w_did_num];        ! Specify the directory
;  1464     1456  5          fib[fib$w_did_seq] = .nam[nam$w_did_seq];
;  1465     1457  5          fib[fib$w_did_rvn] = .nam[nam$w_did_rvn];
;  1466     1458  5
;  1467     1459  5          fib[fib$v_findfid] = true;                       ! Set the findfid bit
;  1468     1460  5
;  1469     1461  5          fib[fib$w_verlimit] = .vrsn_value;               ! And the version limit
;  1470     1462  4          END;
;  1471     1463  4
;  1472     1464  4  !
;  1473     1465  4  ! If any journaling was requested, make those modifications
;  1474     1466  4  !
;  1475     1467  4
;  1476     1468  4      status = 0;                                      ! Flag: journaling info changed.
;  1477     1469  4
;  1478     1470  4      IF .setfile$flags[qual_journal]                  ! /JOURNAL
;  1479     1471  4      THEN
```

```
;  1480     1472  5          BEGIN
;  1481     1473  5          IF .header[fh2$b_struclev] EQL 1         ! If not ODS2
;  1482     1474  5          THEN SIGNAL(set$_notods2,                ! tell the user
;  1483     1475  5                      1,
;  1484     1476  5                      $DESCRIPTOR('/JOURNAL'))
;  1485     1477  5          ELSE
;  1486     1478  6              BEGIN
;  1487     1479  6              !
;  1488     1480  6              ! If any of the RU journal bits are going to be set,
;  1489     1481  6              !    clear the existing header RU bits to avoid conflicts
;  1490     1482  6              !
;  1491     1483  7              IF (.setfile$jflags[jrnl_only_ru] OR .setfile$jflags[jrnl_never_ru]
;  1492     1484  7                                OR .setfile$jflags[jrnl_ru])
;  1493     1485  6              THEN
;  1494     1486  7                  BEGIN
;  1495     1487  7                  header[fh2$v_rujnl] = 0;
;  1496     1488  7                  header[fh2$v_only_ru] = 0;
;  1497     1489  7                  header[fh2$v_never_ru] = 0;
;  1498     1490  6                  END;
;  1499     1491  6
;  1500     1492  6              IF .setfile$jflags[jrnl_specified_ru]
;  1501     1493  6                                           ! RU
;  1502     1494  6              THEN
;  1503     1495  7                  BEGIN
;  1504     1496  7                  header[fh2$v_rujnl] = .setfile$jflags[jrnl_ru];
;  1505     1497  7                  setfile$mflags[misc_mark_file] = 1;
;  1506     1498  7                  status = 1;
;  1507     1499  7                  END
;  1508     1500  6              ELSE IF .setfile$jflags[jrnl_specified_only_ru]
;  1509     1501  6                                           ! RU only
;  1510     1502  6              THEN
;  1511     1503  7                  BEGIN
;  1512     1504  7                  header[fh2$v_only_ru] = .setfile$jflags[jrnl_only_ru];
;  1513     1505  7                  setfile$mflags[misc_mark_file] = 1;
;  1514     1506  7                  status = 1;
;  1515     1507  7                  END
;  1516     1508  6              ELSE IF .setfile$jflags[jrnl_specified_never_ru]
;  1517     1509  6                                           ! RU never
;  1518     1510  6              THEN
;  1519     1511  7                  BEGIN
;  1520     1512  7                  header[fh2$v_never_ru] = .setfile$jflags[jrnl_never_ru];
;  1521     1513  7                  setfile$mflags[misc_mark_file] = 1;
;  1522     1514  7                  status = 1;
;  1523     1515  6                  END;
;  1524     1516  6
;  1525     1517  6              IF .setfile$jflags[jrnl_specified_ai]
;  1526     1518  6                                           ! AI journaling
;  1527     1519  6              THEN
;  1528     1520  7                  BEGIN
;  1529     1521  7                  header[fh2$v_aijnl] = .setfile$jflags[jrnl_ai];
;  1530     1522  7                  setfile$mflags[misc_mark_file] = 1;
;  1531     1523  7                  status = 1;
;  1532     1524  6                  END;
;  1533     1525  6              IF .setfile$jflags[jrnl_specified_at]
;  1534     1526  6                                           ! AT journaling
;  1535     1527  6              THEN
;  1536     1528  7                  BEGIN
```

```
;  1537          1529  7                header[fh2$v_atjnl] = .setfile$jflags[jrnl_at];
;  1538          1530  7                setfile$mflags[misc_mark_file] = 1;
;  1539          1531  7                status = 1;
;  1540          1532  6                END;
;  1541          1533  6            IF .setfile$jflags[jrnl_specified_bi]
;  1542          1534  6                                        ! BI journaling
;  1543          1535  6            THEN
;  1544          1536  7                BEGIN
;  1545          1537  7                header[fh2$v_bijnl] = .setfile$jflags[jrnl_bi];
;  1546          1538  7                setfile$mflags[misc_mark_file] = 1;
;  1547          1539  7                status = 1;
;  1548          1540  6                END;
;  1549          1541  5            END;
;  1550          1542  4        END;
;  1551          1543  4
;  1552          1544  4 !
;  1553          1545  4 ! If there were any journal bits set, show it.
;  1554          1546  4 !
;  1555          1547  4
;  1556          1548  5     IF (.status EQL 1)
;  1557          1549  4     THEN
;  1558          1550  5         BEGIN
;  1559          1551  5         atr[.ptr,atr$w_type] = atr$c_journal;
;  1560          1552  5         atr[.ptr,atr$w_size] = atr$s_journal;
;  1561          1553  5         atr[.ptr,atr$l_addr] = header[fh2$w_journal];
;  1562          1554  5         ptr = .ptr + 1;
;  1563          1555  4         END;
;  1564          1556  4
;  1565          1557  4 !
;  1566          1558  4 ! If an rmsjnlid ACE needs to be added, add it here.
;  1567          1559  4 !
;  1568          1560  4 ! (Only one rmsjnlid ace is needed for journaled file.)
;  1569          1561  4 !
;  1570          1562  4     IF .setfile$mflags[misc_mark_file] AND NOT .setfile$mflags[misc_already_rmsjnlid]
;  1571          1563  4     THEN
;  1572          1564  5         BEGIN
;  1573          1565  5         !
;  1574          1566  5         ! Build JNLID ACE: volume name + file ID + current date/time.
;  1575          1567  5         !
;  1576          1568  5         rmsjnlid_ace[ace$b_size] = id_ace$s_size;
;  1577          1569  5         rmsjnlid_ace[ace$b_type] = ace$c_jnlid;
;  1578          1570  5         rmsjnlid_ace[ace$w_flags] =
;  1579          1571  5                 ace$m_hidden OR ace$m_protected OR ace$m_nopropagate;
;  1580          1572  5
;  1581        P 1573  5         $ITMLST_INIT (ITMLST = item_list, (ITMCOD = dvi$_volnam,
;  1582          1574  5                 BUFADR = label_buffer, BUFSIZ = 12));
;  1583          1575  5
;  1584        P 1576  5         status = $GETDVI(EFN = 1, CHAN = .channel, ITMLST = item_list,
;  1585          1577  5                                    IOSB = iosb);
;  1586          1578  5         IF .status
;  1587          1579  5         THEN
;  1588          1580  6             $WAITFR(EFN = 1)
;  1589          1581  5         ELSE
;  1590          1582  5             SIGNAL( set$_badlogic, 0, .status, 0 );
;  1591          1583  5         IF NOT .iosb[0]
;  1592          1584  5         THEN
;  1593          1585  5             SIGNAL( set$_badlogic, 0, .iosb[0], 0 );
```

```
;  1594          1586  5
;  1595          1587  5          CH$MOVE(12, label_buffer, rmsjnlid_ace[id_ace$t_label]);
;  1596          1588  5
;  1597          1589  5          rmsjnlid_ace[id_ace$w_num] = .nam[nam$w_fid_num];
;  1598          1590  5          rmsjnlid_ace[id_ace$w_seq] = .nam[nam$w_fid_seq];
;  1599          1591  5          rmsjnlid_ace[id_ace$w_rvn] = .nam[nam$w_fid_rvn];
;  1600          1592  5
;  1601          1593  5          $GETTIM( TIMADR = rmsjnlid_ace[id_ace$q_time] );
;  1602          1594  5
;  1603          1595  5          atr[.ptr,atr$w_type] = atr$c_addaclent;
;  1604          1596  5          atr[.ptr,atr$w_size] = id_ace$s_size;
;  1605          1597  5          atr[.ptr,atr$l_addr] = rmsjnlid_ace;
;  1606          1598  5          ptr = .ptr + 1;
;  1607          1599  4          END;
;  1608          1600  4
;  1609          1601  4
;  1610          1602  4  !
;  1611          1603  4  ! Record journals to use in access control list.
;  1612          1604  4  !
;  1613          1605  4
;  1614          1606  4  !
;  1615          1607  4  ! AI Journal Name
;  1616          1608  4  !
;  1617          1609  4
;  1618          1610  4      IF .setfile$jflags[jrnl_ai_name]
;  1619          1611  4      THEN
;  1620          1612  5          BEGIN
;  1621          1613  5          atr[.ptr,atr$w_type] = atr$c_addaclent;
;  1622          1614  5          atr[.ptr,atr$w_size] = 4 + .ai_jnl_desc[dsc$w_length];
;  1623          1615  5          atr[.ptr,atr$l_addr] = ai_jnl_ace;
;  1624          1616  5          ptr = .ptr + 1;
;  1625          1617  5
;  1626          1618  5          ai_jnl_ace[ace$b_size] = 4 + .ai_jnl_desc[dsc$w_length];
;  1627          1619  5          ai_jnl_ace[ace$b_type] = ace$c_aijnl;
;  1628          1620  5          ai_jnl_ace[ace$w_flags] = ace$m_hidden OR ace$m_protected;
;  1629          1621  5
;  1630          1622  5          CH$MOVE (.ai_jnl_desc[dsc$w_length],
;  1631          1623  5                   .ai_jnl_desc[dsc$a_pointer],
;  1632          1624  5                   ai_jnl_ace[ace$t_jnlnam]);
;  1633          1625  4          END;
;  1634          1626  4
;  1635          1627  4  !
;  1636          1628  4  ! AT Journal Name
;  1637          1629  4  !
;  1638          1630  4
;  1639          1631  4      IF .setfile$jflags[jrnl_at_name]
;  1640          1632  4      THEN
;  1641          1633  5          BEGIN
;  1642          1634  5          atr[.ptr,atr$w_type] = atr$c_addaclent;
;  1643          1635  5          atr[.ptr,atr$w_size] = 4 + .at_jnl_desc[dsc$w_length];
;  1644          1636  5          atr[.ptr,atr$l_addr] = at_jnl_ace;
;  1645          1637  5          ptr = .ptr + 1;
;  1646          1638  5
;  1647          1639  5          at_jnl_ace[ace$b_size] = 4 + .at_jnl_desc[dsc$w_length];
;  1648          1640  5          at_jnl_ace[ace$b_type] = ace$c_atjnl;
;  1649          1641  5          at_jnl_ace[ace$w_flags] = ace$m_hidden OR ace$m_protected;
;  1650          1642  5
```

```
;  1651     1643  5          CH$MOVE (.at_jnl_desc[dsc$w_length],
;  1652     1644  5                   .at_jnl_desc[dsc$a_pointer],
;  1653     1645  5                   at_jnl_ace[ace$t_jnlnam]);
;  1654     1646  4          END;
;  1655     1647  4
;  1656     1648  4      !
;  1657     1649  4      ! BI Journal Name
;  1658     1650  4      !
;  1659     1651  4
;  1660     1652  4          IF .setfile$jflags[jrnl_bi_name]
;  1661     1653  4          THEN
;  1662     1654  5              BEGIN
;  1663     1655  5              atr[.ptr,atr$w_type] = atr$c_addaclent;
;  1664     1656  5              atr[.ptr,atr$w_size] = 4 + .bi_jnl_desc[dsc$w_length];
;  1665     1657  5              atr[.ptr,atr$l_addr] = bi_jnl_ace;
;  1666     1658  5              ptr = .ptr + 1;
;  1667     1659  5
;  1668     1660  5              bi_jnl_ace[ace$b_size] = 4 + .bi_jnl_desc[dsc$w_length];
;  1669     1661  5              bi_jnl_ace[ace$b_type] = ace$c_bijnl;
;  1670     1662  5              bi_jnl_ace[ace$w_flags] = ace$m_hidden OR ace$m_protected;
;  1671     1663  5
;  1672     1664  5              CH$MOVE (.bi_jnl_desc[dsc$w_length],
;  1673     1665  5                       .bi_jnl_desc[dsc$a_pointer],
;  1674     1666  5                       bi_jnl_ace[ace$t_jnlnam]);
;  1675     1667  4              END;
;  1676     1668  4
;  1677     1669  4
;  1678     1670  4      !
;  1679     1671  4      ! Write the modifications out to the file header, and close the file.
;  1680     1672  4      !
;  1681     1673  4
;  1682     1674  4          fib[fib$l_aclctx] = 0;                  ! Make sure RMS journaling ACEs go first.
;  1683     1675  4          atr[.ptr,0,0,32,0] = 0;                 ! Put a zero at end of attribute list
;  1684     1676  4
;  1685     1677  5          IF (    .ptr NEQ 0                      ! If an attribute was changed
;  1686     1678  5              OR .setfile$flags[qual_trunc]       ! Or the file should be truncated
;  1687     1679  5              OR .setfile$flags[qual_vrsn])       ! Or the version_limit set
;  1688     1680  4          THEN
;  1689     1681  5              BEGIN
;  1690     1682  5              LOCAL   RES_DESC : $BBLOCK [8],
;  1691     1683  5                      RES_BUF : $BBLOCK [512],
;  1692     1684  5                      RES_LEN;
;  1693     1685  5              CH$FILL (0, 8, RES_DESC);
;  1694     1686  5              RES_DESC[DSC$W_LENGTH] = 512;
;  1695     1687  5              RES_DESC[DSC$A_POINTER] = RES_BUF;
;  1696  P  1688  5              status = $QIOW (CHAN = .channel,        ! Make the modifications
;  1697  P  1689  5                              FUNC = IO$_MODIFY,
;  1698  P  1690  5                              IOSB = iosb,
;  1699  P  1691  5                              P1 = desc,
;  1700  P  1692  5                              P3 = RES_LEN,
;  1701  P  1693  5                              P4 = RES_DESC,
;  1702     1694  5                              P5 = atr);
;  1703     1695  5              IF .status THEN status = .iosb[0];
;  1704     1696  5              IF NOT .status
;  1705     1697  5              THEN file_error(set$_writeerr,.status,.fab)     ! If the modify failed, tell user
;  1706     1698  5              ELSE
;  1707     1699  5                  IF .setfile$flags[qual_log]         ! If /LOG, tell user
```

```
;  1708     1700  5            THEN SIGNAL(set$_modified,1,conf_desc);
;  1709     1701  4        END;
;  1710     1702  4
;  1711     1703  4    !
;  1712     1704  4    ! Now to close the file.  Don't bother to check the status, since the
;  1713     1705  4    ! modifications got made correctly.
;  1714     1706  4    !
;  1715   P 1707  4        $QIOW( CHAN = .channel,                         ! Deaccess the file
;  1716   P 1708  4               FUNC = IO$_DEACCESS,
;  1717   P 1709  4               IOSB = iosb,
;  1718     1710  4               P1 = desc);
;  1719     1711  3        END;
;  1720     1712  2    END;
;  1721     1713  2
;  1722     1714  2    !
;  1723     1715  2    ! If /REMOVE or /ENTER was specified, process it
;  1724     1716  2    !
;  1725     1717  2    IF (.setfile$flags[qual_remove] OR .setfile$flags[qual_enter])
;  1726     1718  2    THEN
;  1727     1719  3        BEGIN
;  1728     1720  3
;  1729     1721  3    !
;  1730     1722  3    ! Set up the FIB appropriately
;  1731     1723  3    !
;  1732     1724  3
;  1733     1725  3        fib[fib$w_did_num] = .nam[nam$w_did_num];   ! Put in the directory ID
;  1734     1726  3        fib[fib$w_did_seq] = .nam[nam$w_did_seq];
;  1735     1727  3        fib[fib$w_did_rvn] = .nam[nam$w_did_rvn];
;  1736     1728  3
;  1737     1729  3    !
;  1738     1730  3    ! If /REMOVE was specified, remove the directory entry
;  1739     1731  3    !
;  1740     1732  3        IF .setfile$flags[qual_remove]
;  1741     1733  3        THEN
;  1742     1734  4            BEGIN
;  1743     1735  4    !
;  1744     1736  4    ! Check to see if an explicit or wild version number was specified.
;  1745     1737  4    ! If not, exit with an error.
;  1746     1738  4    !
;  1747     1739  5            IF NOT (.nam[nam$v_exp_ver] OR
;  1748     1740  5                    .nam[nam$v_wild_ver])
;  1749     1741  4            THEN SIGNAL(set$_remerr,
;  1750     1742  4                        1,
;  1751     1743  4                        conf_desc,
;  1752     1744  4                        set$_delver)
;  1753     1745  4    !
;  1754     1746  4    ! If /CONFIRM was set by the user, then interrogate him to see
;  1755     1747  4    ! if the directory entry is to be removed.
;  1756     1748  4    !
;  1757     1749  4            ELSE
;  1758     1750  4            IF
;  1759     1751  5                BEGIN
;  1760     1752  5                IF .setfile$flags[qual_quit_rem]
;  1761     1753  5                OR NOT .setfile$flags[qual_confirm]
;  1762     1754  5                THEN true
;  1763     1755  5                ELSE
;  1764     1756  6                    BEGIN
```

```
;  1765     1757  6                status = lib$confirm_act(%ASCID 'Remove directory entry for !AS? [N]: ',
;  1766     1758  6                                         %REF(conf_desc));
;  1767     1759  6                IF NOT .status
;  1768     1760  6                THEN
;  1769     1761  7                    BEGIN
;  1770     1762  7                    IF .status EQL lib$_quipro
;  1771     1763  8                    THEN (setfile$flags[qual_quit] = 1; RETURN true)
;  1772     1764  7                    ELSE IF .status EQL lib$_quiconact
;  1773     1765  8                    THEN (setfile$flags[qual_quit_rem] = 1; status = 1)
;  1774     1766  7                    ELSE IF .status NEQ lib$_negans
;  1775     1767  7                    THEN SIGNAL(set$_writeerr, 1, conf_desc, .status);
;  1776     1768  6                    END;
;  1777     1769  6                .status
;  1778     1770  6                END
;  1779     1771  5            END
;  1780     1772  4        THEN
;  1781     1773  5            BEGIN
;  1782     1774  5
;  1783     1775  5            fib[fib$w_fid_num] = 0;                     ! Clear the File ID
;  1784     1776  5            fib[fib$w_fid_seq] = 0;
;  1785     1777  5            fib[fib$w_fid_rvn] = 0;
;  1786     1778  5
;  1787     1779  5
;  1788     1780  5        !
;  1789     1781  5        ! Isolate the file portion of the resultant file string
;  1790     1782  5        !
;  1791     1783  5            file_name[0] = .nam[nam$b_name]
;  1792     1784  5                         + .nam[nam$b_type]
;  1793     1785  5                         + .nam[nam$b_ver];
;  1794     1786  5            file_name[1] = .nam[nam$l_name];
;  1795     1787  5
;  1796     1788  5        !
;  1797     1789  5        ! Issue the QIO to remove the directory entry
;  1798     1790  5        !
;  1799   P 1791  5            status = $QIOW( CHAN = .channel,
;  1800   P 1792  5                            FUNC = IO$_DELETE,
;  1801   P 1793  5                            IOSB = iosb,
;  1802   P 1794  5                            P1 = desc,
;  1803     1795  5                            P2 = file_name);
;  1804     1796  5            IF .status THEN status = .iosb[0];
;  1805     1797  5            IF NOT .status
;  1806     1798  5            THEN SIGNAL(set$_writeerr,          ! Error writing
;  1807     1799  5                        1,
;  1808     1800  5                        file_name,                      ! to this file
;  1809     1801  5                        .status)                        ! for this reason
;  1810     1802  5            ELSE
;  1811     1803  5                IF .setfile$flags[qual_log]             ! If /LOG, tell user
;  1812     1804  5                THEN SIGNAL(set$_removed,1,conf_desc);
;  1813     1805  4            END;
;  1814     1806  4        END
;  1815     1807  4
;  1816     1808  4    ! IF /ENTER, enter the name in the directory
;  1817     1809  4    !
;  1818     1810  3        ELSE
;  1819     1811  4            BEGIN
;  1820     1812  4            LOCAL
;  1821     1813  4                new_name : VECTOR[2],               ! Place to put new filespec
```

```
;  1822          1814  4              new_fab : $BBLOCK[fab$c_bln],       ! Temp output fab
;  1823          1815  4              new_nam : $BBLOCK[nam$c_bln],       ! Temp output name block
;  1824          1816  4              new_nam2 : $BBLOCK[nam$c_bln],      !  another temp nam block
;  1825          1817  4              new_desc : $BBLOCK[dsc$c_s_bln],    ! Descriptor for new name
;  1826          1818  4              new_nam_exp : VECTOR[nam$c_maxrss,BYTE], ! Expanded string
;  1827          1819  4              new_nam_exp2 : VECTOR[nam$c_maxrss,BYTE];! Expanded string2
;  1828          1820  4
;  1829          1821  4      !
;  1830          1822  4      ! Initialize the fab and name block, using the original file name block
;  1831          1823  4      ! as the related file.
;  1832        P 1824  4      !
;  1833        P 1825  4            $FAB_INIT ( FAB = new_fab,
;  1834        P 1826  4                        NAM = new_nam,
;  1835          1827  4                        FNA = .file_name[1],
;  1836        P 1828  4                        FNS = .file_name[0]);
;  1837        P 1829  4            $NAM_INIT ( NAM = new_nam,
;  1838        P 1830  4                        RLF = nam,
;  1839          1831  4                        ESA = new_nam_exp,
;  1840          1832  4                        ESS = nam$c_maxrss);
;  1841          1833  4      !
;  1842          1834  4      ! If the original file specification had a wildcard version number, then
;  1843          1835  4      ! use one here.
;  1844          1836  5      !
;  1845          1837  4            IF (.nam[nam$v_wild_ver])
;  1846          1838  5            THEN
;  1847          1839  5                BEGIN
;  1848          1840  5                new_fab[fab$l_dna] = UPLIT(';*');
;  1849          1841  5                new_fab[fab$b_dns] = %CHARCOUNT(';*');
;  1850          1842  4                END
;  1851          1843  5            ELSE
;  1852          1844  5                BEGIN
;  1853          1845  5                new_fab[fab$l_dna] = 0;
;  1854          1846  4                new_fab[fab$b_dns] = 0;
;  1855          1847  4                END;
;  1856          1848  4      !
;  1857          1849  4      ! Parse once, with the OFP bit off, to fill in all the fields
;  1858          1850  4      ! from the original name block
;  1859          1851  4      !
;  1860          1852  4            status = $PARSE (FAB = new_fab);
;  1861          1853  4            CH$MOVE(nam$c_bln,new_nam,new_nam2);
;  1862          1854  4            parse_null_string(new_fab);
;  1863          1855  4            IF NOT .status
;  1864          1856  5            THEN
;  1865          1857  5                BEGIN
;  1866          1858  5                SIGNAL_STOP(set$_enterr,
;  1867          1859  5                            2,
;  1868          1860  5                            conf_desc,
;  1869          1861  5                            file_name,
;  1870          1862  5                            .status);
;  1871          1863  4                RETURN true;
;  1872          1864  4                END;
;  1873          1865  4      !
;  1874          1866  4      ! Now parse with OFP set, to obtain the final file name
;  1875          1867  4      !
;  1876        P 1868  4            $FAB_INIT(  FAB = new_fab,
;  1877        P 1869  4                        NAM = new_nam,
;  1878        P 1870  4                        FNA = .new_nam2[nam$l_esa],
```

```
;  1879        P 1871  4                       FNS = .new_nam2[nam$b_esl],
;  1880          1872  4                       FOP = ofp);
;  1881        P 1873  4           $NAM_INIT(  NAM = new_nam,
;  1882        P 1874  4                       RLF = nam,
;  1883        P 1875  4                       ESA = new_nam_exp2,
;  1884          1876  4                       ESS = nam$c_maxrss);
;  1885          1877  4
;  1886          1878  4           status = $PARSE (FAB = new_fab);
;  1887          1879  4           CH$MOVE(nam$c_bln,new_nam,new_nam2);
;  1888          1880  4           parse_null_string(new_fab);
;  1889          1881  4           IF NOT .status
;  1890          1882  4           THEN
;  1891          1883  5               BEGIN
;  1892          1884  5               SIGNAL(set$_enterr,                 ! Error entering
;  1893          1885  5                      2,
;  1894          1886  5                      conf_desc,                       ! This file
;  1895          1887  5                      file_name,                       ! as this file
;  1896          1888  5                      .status);                        ! Not on same device
;  1897          1889  5               RETURN true;
;  1898          1890  4               END;
;  1899          1891  4
;  1900          1892  4 !
;  1901          1893  4 ! Get the full file name
;  1902          1894  4 !
;  1903          1895  4           new_name[0] = .new_nam2[nam$b_esl];
;  1904          1896  4           new_name[1] = .new_nam2[nam$l_esa];
;  1905          1897  4
;  1906          1898  4 !
;  1907          1899  4 ! Find the actual file name
;  1908          1900  4 !
;  1909          1901  4           new_desc[dsc$w_length] = .new_nam2[nam$b_name]
;  1910          1902  4                                  + .new_nam2[nam$b_type]
;  1911          1903  4                                  + .new_nam2[nam$b_ver];
;  1912          1904  4           new_desc[dsc$a_pointer] = .new_nam2[nam$l_name];
;  1913          1905  4
;  1914          1906  4 !
;  1915          1907  4 ! Put in the file ID of the target directory
;  1916          1908  4 !
;  1917          1909  4           fib[fib$w_did_num] = .new_nam2[nam$w_did_num];
;  1918          1910  4           fib[fib$w_did_seq] = .new_nam2[nam$w_did_seq];
;  1919          1911  4           fib[fib$w_did_rvn] = .new_nam2[nam$w_did_rvn];
;  1920          1912  4 !
;  1921          1913  4 ! Check to see that the enter request is for the same device.  If not,
;  1922          1914  4 ! signal an error.  This is done by comparing the DVI field of the RMS
;  1923          1915  4 ! name blocks.
;  1924          1916  4 !
;  1925          1917  4
;  1926          1918  4           IF CH$NEQ( .(nam[nam$t_dvi])<0,8>, nam[nam$t_dvi]+1,
;  1927          1919  4                      .(new_nam2[nam$t_dvi])<0,8>, new_nam2[nam$t_dvi]+1, 0)
;  1928          1920  4           THEN SIGNAL(set$_enterr,                    ! Error entering
;  1929          1921  4                       2,
;  1930          1922  4                       conf_desc,                      ! This file
;  1931          1923  4                       new_name,                       ! as this file
;  1932          1924  4                       RMS$_DEV)                       ! Not on same device
;  1933          1925  4           ELSE
;  1934          1926  4 !
;  1935          1927  4 ! If /CONFIRM was set by the user, then interrogate him to see
```

```
! if the file is to be entered in a directory.
!
        IF
            BEGIN
            IF .setfile$flags[qual_quit_ent]
            OR NOT .setfile$flags[qual_confirm]
            THEN true
            ELSE
                BEGIN
                LOCAL
                    arglist : VECTOR[2];
                arglist[0] = conf_desc;
                arglist[1] = new_name;
                status = lib$confirm_act(%ASCID 'Enter !AS as !AS? [N]: ',
                                         arglist);
                IF NOT .status
                THEN
                    BEGIN
                    IF .status EQL lib$_quipro
                    THEN (setfile$flags[qual_quit] = 1; RETURN true)
                    ELSE IF .status EQL lib$_quiconact
                    THEN (setfile$flags[qual_quit_ent] = 1; status = 1)
                    ELSE IF .status NEQ lib$_negans
                    THEN SIGNAL(set$_writeerr, 1, conf_desc, .status);
                    END;
                .status
                END
            END
        THEN
            BEGIN
!
! Issue the QIO
!

            status = $QIOW( CHAN = .channel,     ! Enter the new name
                            FUNC = IO$_CREATE,
                            IOSB = iosb,
                            P1 = desc,
                            P2 = new_desc);
            IF .status THEN status = .iosb[0];
            IF NOT .status
            THEN SIGNAL(set$_enterr,              ! Error enterring
                        2,
                        conf_desc,                ! this file
                        new_name,                 ! as this file
                        .status)                  ! for this reason
            ELSE
                IF .setfile$flags[qual_log]       ! If /LOG, tell user
                THEN SIGNAL(set$_entered,2,conf_desc,new_name);
            END;
        END;                                      ! End of /ENTER block
    END;                                          ! End of modify block
!
! If /UNLOCK was specified by the user
!
    IF .setfile$flags[qual_unlock]
    THEN
```

```
;  1993          1985  3          IF NOT (status = unlock_action(.fab))
;  1994          1986  3          THEN
;  1995          1987  3              SIGNAL(set$_unlockerr,1,conf_desc,.status);
;  1996          1988  3
;  1997          1989  2      !
;  1998          1990  2      ! If /PROTECTION was specified by the user
;  1999          1991  2      !
;  2000          1992  2          IF .setfile$flags[qual_protection]
;  2001          1993  2          THEN
;  2002          1994  3              IF NOT (status = setpro_action(.fab))
;  2003          1995  3              THEN
;  2004          1996  4                  SIGNAL(set$_proerr,1,conf_desc,.status);
;  2005          1997  3
;  2006          1998  2          FI;
;  2007          1999  2      !
;  2008          2000  2      ! Deassign the channel
;  2009          2001  2      !
;  2010          2002  2      IF NOT (status = $DASSGN(CHAN = .channel))
;  2011          2003  2      THEN file_error(set$_closeerr,.status,.fab);          ! If deassign failed, say so
;  2012          2004  2
;  2013          2005  2      RETURN true;                                  ! Continue processing other files
;  2014          2006  1      END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

53 41 21 20 65 6C 69 66 20 79 66 69 64 6F 4D 003A0 P.ADN:   .ASCII  \Modify file !AS? [N] : \<0>                    ;
                  00 20 3A 20 5D 4E 5B 20 3F 003AF
                                          010E0017 003B8 P.ADM:   .LONG   17694743                                  ;
                                          00000000' 003BC         .ADDRESS P.ADN                                    ;
                                    45 53 41 52 45 2F 003C0 P.ADP:   .ASCII  \/ERASE\                               ;
                                                      003C6         .BLKB   2                                         ;
                                          00000006  003C8 P.ADO:   .LONG   6                                        ;
                                          00000000' 003CC         .ADDRESS P.ADP                                    ;
                              45 53 41 52 45 4F 4E 2F 003D0 P.ADR:   .ASCII  \/NOERASE\                             ;
                                          00000008  003D8 P.ADQ:   .LONG   8                                        ;
                                          00000000' 003DC         .ADDRESS P.ADR                                    ;
                        59 52 4F 54 43 45 52 49 44 4F 4E 2F 003E0 P.ADT:   .ASCII  \/NODIRECTORY\                   ;
                                          0000000C  003EC P.ADS:   .LONG   12                                       ;
                                          00000000' 003F0         .ADDRESS P.ADT                                    ;
                              4C 41 4E 52 55 4F 4A 2F 003F4 P.ADV:   .ASCII  \/JOURNAL\                           ;
                                          00000008  003FC P.ADU:   .LONG   8                                        ;
                                          00000000' 00400         .ADDRESS P.ADV                                    ;
72 6F 7A 63 65 72 69 64 20 65 76 6F 6D 65 52 00404 P.ADX:   .ASCII  \Remove directory entry for !AS? [N]: \<0>       ;
53 41 21 20 72 6F 66 20 79 72 74 6E 65 20 79 00413
                  00 20 3A 5D 4E 5B 20 3F 00422
                        00 00             0042A         .ASCII  <0><0>                                            ;
                                          010E0025  0042C P.ADW:   .LONG   17694757                                 ;
                                          00000000' 00430         .ADDRESS P.ADX                                    ;
                                    00 00 2A 3B 00434 P.ADY:   .ASCII  \:*\<0><0>                                   ;
41 21 20 73 61 20 53 41 21 20 72 65 74 6E 45 00438 P.AEA:   .ASCII  \Enter !AS as !AS? [N]: \<0>                    ;
                  00 20 3A 5D 4E 5B 20 3F 53 00447
                                          010E0017  00450 P.ADZ:   .LONG   17694743                                 ;
                                          00000000' 00454         .ADDRESS P.AEA                                    ;

                                                                .PSECT  $OWN$,NOEXE,2
```

```
                                                     00004 RMS$JNLID_ACE:
                                                                 .BLKB    32
                                                     00024 OLD_DID_NUM:
                                                                 .BLKB    2
                                                     00026 OLD_DID_SEQ:
                                                                 .BLKB    2
                                                     00028 OLD_DID_RVN:
                                                                 .BLKB    2

                                                                 .EXTRN   SYS$ASSIGN, SYS$QIOW
                                                                 .EXTRN   SYS$DASSGN, SYS$GETDVI
                                                                 .EXTRN   SYS$WAITFR, SYS$GETTIM

                                                                 .PSECT   $CODE$,NOWRT,2

                                                OFFC 00000 SET_ATTRIBUTES:
                                                                 .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11           ; 1016
                           5B 00000000'   EF   9E 00002          MOVAB    SETFILE$FLAGS, R11                             ;
                           5E     F9B8    CE   9E 00009          MOVAB    -1608(SP), SP                                  ;
                           5A       04    AC   D0 0000E          MOVL     FAB, R10                                       ; 1050
                           57       28    AA   D0 00012          MOVL     40(R10), R7                                    ;
           47      02      AB             06   E0 00016          BBS      #6, SETFILE$FLAGS+2, 2$                        ; 1063
                 0190      CB       04    A7   D0 0001B          MOVL     4(R7), CONF_DESC+4                             ; 1069
                 018C      CB       03    A7   9B 00021          MOVZBW   3(R7), CONF_DESC                               ; 1070
                                    7E    7C 00027               CLRQ     -(SP)                                          ; 1072
                                    7E    D4 00029               CLRL     -(SP)                                          ;
                                  018C    CB   9F 0002B          PUSHAB   CONF_DESC                                      ;
                                    7E    D4 0002F               CLRL     -(SP)                                          ;
                              00000000'   EF   9F 00031          PUSHAB   CONTEXT                                        ;
              00000000G    00             06   FB 00037          CALLS    #6, LIB$QUAL_FILE_MATCH                        ;
                           58             50   D0 0003E          MOVL     R0, STATUS                                     ;
                           21             58   E8 00041          BLBS     STATUS, 3$                                     ; 1078
              00000000G    8F             58   D1 00044          CMPL     STATUS, #LIB$_FILFAIMAT                        ; 1081
                                          15   13 0004B          BEQL     2$                                             ;
                                          58   DD 0004D          PUSHL    STATUS                                         ; 1085
                                  018C    CB   9F 0004F          PUSHAB   CONF_DESC                                      ; 1082
                                          01   DD 00053          PUSHL    #1                                             ;
                              0077109A    8F   DD 00055          PUSHL    #7803034                                       ;
              00000000G    00             04   FB 0005B 1$:      CALLS    #4, LIB$SIGNAL                                 ;
                                      0B9B     31 00062 2$:      BRW      102$                                           ; 1086
                   90      AD       39    A7   9B 00065 3$:      MOVZBW   57(R7), DESC                                   ; 1093
                   94      AD       44    A7   D0 0006A          MOVL     68(R7), DESC+4                                 ; 1094
                                          7E   7C 0006F          CLRQ     -(SP)                                          ; 1097
                                    0C    AE   9F 00071          PUSHAB   CHANNEL                                        ;
                                    90    AD   9F 00074          PUSHAB   DESC                                           ;
              00000000G    00             04   FB 00077          CALLS    #4, SYS$ASSIGN                                 ;
                           58             50   D0 0007E          MOVL     R0, STATUS                                     ;
                           0D             58   E8 00081          BLBS     STATUS, 4$                                     ;
                                  0500    8F   BB 00084          PUSHR    #^M<R8,R10>                                    ; 1101
                              0077109A    8F   DD 00088          PUSHL    #7803034                                       ; 1100
                                      0B68     31 0008E          BRW      101$                                           ;
                   90      AD       40    8F   9B 00091 4$:      MOVZBW   #64, DESC                                      ; 1108
                   94      AD     FF50    CD   9E 00096          MOVAB    FIB, DESC+4                                    ; 1109
 0040     8F             00         6E    00   2C 0009C          MOVC5    #0, (SP), #0, #64, FIB                         ; 1111
                                  FF50    CD      000A3                                                                  ;
                 FF50      CD     0501    8F   3C 000A6          MOVZWL   #1281, FIB                                     ; 1114
```

```
                FF54   CD          24   A7   D0 000AD          MOVL    36(R7), FIB+4                                   ; 1117
                FF58   CD          28   A7   B0 000B3          MOVW    40(R7), FIB+8                                   ; 1119
          3E           6B               01   E0 000B9          BBS     #1, SETFILE$FLAGS, 5$                           ; 1126
          3A           6B               02   E0 000BD          BBS     #2, SETFILE$FLAGS, 5$                           ; 1127
          36           6B               04   E0 000C1          BBS     #4, SETFILE$FLAGS, 5$                           ; 1128
          32           6B               05   E0 000C5          BBS     #5, SETFILE$FLAGS, 5$                           ; 1129
          2E           6B               06   E0 000C9          BBS     #6, SETFILE$FLAGS, 5$                           ; 1130
                                        6B   95 000CD          TSTB    SETFILE$FLAGS                                   ; 1131
                                        2A   19 000CF          BLSS    5$
                       26         01    AB   E8 000D1          BLBS    SETFILE$FLAGS+1, 5$                             ; 1132
          21   01      AB               01   E0 000D5          BBS     #1, SETFILE$FLAGS+1, 5$                         ; 1133
          1C   01      AB               02   E0 000DA          BBS     #2, SETFILE$FLAGS+1, 5$                         ; 1134
          17   01      AB               03   E0 000DF          BBS     #3, SETFILE$FLAGS+1, 5$                         ; 1135
          12   01      AB               05   E0 000E4          BBS     #5, SETFILE$FLAGS+1, 5$                         ; 1136
          0D   01      AB               06   E0 000E9          BBS     #6, SETFILE$FLAGS+1, 5$                         ; 1137
          08   02      AB               01   E0 000EE          BBS     #1, SETFILE$FLAGS+2, 5$                         ; 1138
          03   02      AB               03   E0 000F3          BBS     #3, SETFILE$FLAGS+2, 5$                         ; 1139
                                  0700   31   000F8          BRW     72$
                          02      AB     95   000FB 5$:       TSTB    SETFILE$FLAGS+2                                 ; 1142
                                        5E   19 000FE          BLSS    9$
          5A           6B               03   E1 00100          BBC     #3, SETFILE$FLAGS, 9$                           ; 1143
                       6E       018C    CB   9E 00104          MOVAB   CONF_DESC, (SP)                                 ; 1148
                                        5E   DD 00109          PUSHL   SP
                            00000000'   EF   9F 0010B          PUSHAB  P.ADM                                           ; 1147
          00000000G    00               02   FB 00111          CALLS   #2, LIB$CONFIRM_ACT
                       58               50   D0 00118          MOVL    R0, STATUS
                       40               58   E8 0011B          BLBS    STATUS, 9$                                      ; 1149
          00000000G    8F               58   D1 0011E          CMPL    STATUS, #LIB$_QUIPRO                            ; 1152
                                        03   12 00125          BNEQ    6$
                                  09C3  31   00127          BRW     89$
          00000000G    8F               58   D1 0012A 6$:      CMPL    STATUS, #LIB$_QUICONACT                         ; 1154
                                        0A   12 00131          BNEQ    7$
                02     AB         80    8F   88 00133          BISB2   #128, SETFILE$FLAGS+2                           ; 1155
                       58               01   D0 00138          MOVL    #1, STATUS
                                        1E   11 0013B          BRB     8$
          00000000G    8F               58   D1 0013D 7$:      CMPL    STATUS, #LIB$_NEGANS                            ; 1156
                                        15   13 00144          BEQL    8$
                                        58   DD 00146          PUSHL   STATUS                                          ; 1157
                                  018C  CB   9F 00148          PUSHAB  CONF_DESC
                                        01   DD 0014C          PUSHL   #1
                            00000000G   8F   DD 0014E          PUSHL   #SET$_WRITEERR
          00000000G    00               04   FB 00154          CALLS   #4, LIB$SIGNAL
                       70               58   E9 0015B 8$:      BLBC    STATUS, 11$                                     ; 1159
                98     AD  000A0200     8F   D0 0015E 9$:      MOVL    #655872, ATR                                    ; 1166
                9C     AD        FD50   CD   9E 00166          MOVAB   HEADER, ATR+4                                   ; 1167
                A0     AD  00230020     8F   D0 0016C          MOVL    #2293792, ATR+8                                 ; 1169
                A4     AD  00000000'    EF   9E 00174          MOVAB   RMSJNLID_ACE, ATR+12                            ; 1170
                               A8        AD   D4 0017C          CLRL    ATR+16                                          ; 1171
          00000000'    EF  06000820     8F   D0 0017F          MOVL    #100665376, RMSJNLID_ACE                        ; 1173
                                        7E   D4 0018A          CLRL    -(SP)                                           ; 1181
                          98            AD   9F 0018C          PUSHAB  ATR
                                        7E   7C 0018F          CLRQ    -(SP)
                                        7E   D4 00191          CLRL    -(SP)
                          90            AD   9F 00193          PUSHAB  DESC
                                        7E   7C 00196          CLRQ    -(SP)
                        FCF0            CD   9F 00198          PUSHAB  IOSB
                    7E        72        8F   9A 0019C          MOVZBL  #114, -(SP)
```

```
                                59      2C   AE  3C 001A0          MOVZWL  CHANNEL, R9                                    ;
                                             59  DD 001A4          PUSHL   R9                                           ;
                                             7E  D4 001A6          CLRL    -(SP)                                        ;
                00000000G       00           0C  FB 001A8          CALLS   #12, SYS$QIOW                                ;
                                58           50  D0 001AF          MOVL    R0, STATUS                                   ;
                                08           58  E9 001B2          BLBC    STATUS, 10$                                  ; 1182
                                58      FCF0 CD  3C 001B5          MOVZWL  IOSB, STATUS                                 ;
                                14           58  E8 001BA          BLBS    STATUS, 12$                                  ; 1183
                                        0500 8F  BB 001BD 10$:     PUSHR   #^M<R8,R10>                                  ; 1184
                                00000000G    8F  DD 001C1          PUSHL   #SET$_READERR                                ;
                00000000V       EF           03  FB 001C7          CALLS   #3, FILE_ERROR                               ;
                                          062A  31 001CE 11$:      BRW     72$                                          ;
                                01      FD57 CD  91 001D1 12$:     CMPB    HEADER+7, #1                                 ; 1191
                                             08  12 001D6          BNEQ    13$                                          ;
FD64    CD      FD5E    CD                   20  28 001D8          MOVC3   #32, HEADER+14, HEADER+20                    ; 1192
                                00000000'    EF  95 001E0 13$:     TSTB    RMSJNLID_ACE+1                               ; 1198
                                             04  13 001E6          BEQL    14$                                          ;
                        0C      AB           04  88 001E8          BISB2   #4, SETFILE$MFLAGS                           ; 1200
                                             56  D4 001EC 14$:     CLRL    PTR                                          ; 1207
                                             58  D4 001EE          CLRL    STATUS                                       ; 1212
                08              6B           01  E1 001F0          BBC     #1, SETFILE$FLAGS, 15$                       ; 1214
                        FD84    CD           02  8A 001F4          BICB2   #2, HEADER+52                                ; 1217
                                58           01  D0 001F9          MOVL    #1, STATUS                                   ; 1218
                08              6B           02  E1 001FC 15$:     BBC     #2, SETFILE$FLAGS, 16$                       ; 1220
                        FD84    CD           02  88 00200          BISB2   #2, HEADER+52                                ; 1223
                                58           01  D0 00205          MOVL    #1, STATUS                                   ; 1224
                26              6B           06  E1 00208 16$:     BBC     #6, SETFILE$FLAGS, 18$                       ; 1227
                                01      FD57 CD  91 0020C          CMPB    HEADER+7, #1                                 ; 1230
                                             17  12 00211          BNEQ    17$                                          ;
                                00000000'    EF  9F 00213          PUSHAB  P.ADD                                        ; 1233
                                             01  DD 00219          PUSHL   #1                                           ; 1231
                                00000000G    8F  DD 0021B          PUSHL   #SET$_NOTODS2                                ;
        00000000G       00                   03  FB 00221          CALLS   #3, LIB$SIGNAL                               ;
                                             08  11 00228          BRB     18$                                          ;
                FD86    CD                   02  88 0022A 17$:     BISB2   #2, HEADER+54                                ; 1236
                                58           01  D0 0022F          MOVL    #1, STATUS                                   ; 1237
                                             6B  95 00232 18$:     TSTB    SETFILE$FLAGS                                ; 1240
                                             26  18 00234          BGEQ    20$                                          ;
                                01      FD57 CD  91 00236          CMPB    HEADER+7, #1                                 ; 1243
                                             17  12 0023B          BNEQ    19$                                          ;
                                00000000'    EF  9F 0023D          PUSHAB  P.ADQ                                        ; 1246
                                             01  DD 00243          PUSHL   #1                                           ; 1244
                                00000000G    8F  DD 00245          PUSHL   #SET$_NOTODS2                                ;
        00000000G       00                   03  FB 0024B          CALLS   #3, LIB$SIGNAL                               ;
                                             08  11 00252          BRB     20$                                          ;
                FD86    CD                   02  8A 00254 19$:     BICB2   #2, HEADER+54                                ; 1249
                                58           01  D0 00259          MOVL    #1, STATUS                                   ; 1250
        2B              6B                   04  E1 0025C 20$:     BBC     #4, SETFILE$FLAGS, 25$                       ; 1254
        05      04      AB                   01  E1 00260          BBC     #1, SETFILE$DFLAGS, 21$                      ; 1257
                FD84    CD                   08  88 00265          BISB2   #8, HEADER+52                                ;
        05      04      AB                   03  E1 0026A 21$:     BBC     #3, SETFILE$DFLAGS, 22$                      ; 1258
                FD84    CD                   08  8A 0026F          BICB2   #8, HEADER+52                                ;
        05      04      AB                   02  E1 00274 22$:     BBC     #2, SETFILE$DFLAGS, 23$                      ; 1259
                FD84    CD                   10  88 00279          BISB2   #16, HEADER+52                               ;
        05      04      AB                   04  E1 0027E 23$:     BBC     #4, SETFILE$DFLAGS, 24$                      ; 1260
                FD84    CD                   10  8A 00283          BICB2   #16, HEADER+52                               ;
                                58           01  D0 00288 24$:     MOVL    #1, STATUS                                   ; 1261
```

```
      26        01  AB           05 E1 0028B 25$:    BBC     #5, SETFILE$FLAGS+1, 27$              ; 1264
                    01     FD57  CD 91 00290         CMPB    HEADER+7, #1                          ; 1267
                                 17 12 00295         BNEQ    26$                                   ;
                      00000000'  EF 9F 00297         PUSHAB  P.ADS                                 ; 1270
                                 01 DD 0029D         PUSHL   #1                                    ; 1268
                      00000000G  8F DD 0029F         PUSHL   #SET$_NOTODS2                         ;
          00000000G  00          03 FB 002A5         CALLS   #3, LIB$SIGNAL                        ;
                                 08 11 002AC         BRB     27$                                   ;
            FD85  CD             20 8A 002AE 26$:    BICB2   #32, HEADER+53                        ; 1273
                  58             01 D0 002B3         MOVL    #1, STATUS                            ; 1274
                  18             58 E9 002B6 27$:    BLBC    STATUS, 28$                           ; 1281
                          9A AD46 7F 002B9           PUSHAQ  ATR+2[PTR]                            ; 1284
                  9E             03 B0 002BD         MOVW    #3, @(SP)+                            ;
                          98 AD46 7F 002C0           PUSHAQ  ATR[PTR]                              ; 1285
                  9E             04 B0 002C4         MOVW    #4, @(SP)+                            ;
                          9C AD46 7F 002C7           PUSHAQ  ATR+4[PTR]                            ; 1286
                  9E       FD84  CD 9E 002CB         MOVAB   HEADER+52, @(SP)+                     ;
                                 56 D6 002D0         INCL    PTR                                   ; 1287
                                 58 D4 002D2         CLRL    STATUS                                ; 1288
      09        01  AB           01 E1 002D4 28$:    BBC     #1, SETFILE$FLAGS+1, 29$              ; 1295
            FD76  CD        20   AB B0 002D9         MOVW    EXTE_VALUE, RECATTR+18                ; 1298
                  58             01 D0 002DF         MOVL    #1, STATUS                            ; 1299
      3C          68             05 E1 002E2 29$:    BBC     #5, SETFILE$FLAGS, 31$                ; 1306
                  01        24   A7 B1 002E6         CMPW    36(R7), #1                            ; 1309
                                 25 12 002EA         BNEQ    30$                                   ;
                  01        26   A7 B1 002EC         CMPW    38(R7), #1                            ; 1310
                                 1F 12 002F0         BNEQ    30$                                   ;
                            29   A7 95 002F2         TSTB    41(R7)                                ; 1311
                                 1A 12 002F5         BNEQ    30$                                   ;
                  7E      0800   8F 3C 002F7         MOVZWL  #2048, -(SP)                          ; 1312
                          018C   CB 9F 002FC         PUSHAB  CONF_DESC                             ;
                                 01 DD 00300         PUSHL   #1                                    ;
                      000000006  8F DD 00302         PUSHL   #SET$_WRITEERR                        ;
          00000000G  00          04 FB 00308         CALLS   #4, LIB$SIGNAL                        ;
                                 11 11 0030F         BRB     31$                                   ;
            FD6C  CD       FD68  CD D0 00311 30$:    MOVL    RECATTR+4, RECATTR+8                  ; 1318
            FD70  CD       0200  8F B0 00318         MOVW    #512, RECATTR+12                      ; 1319
                  58             01 D0 0031F         MOVL    #1, STATUS                            ; 1320
      09        01  AB           02 E1 00322 31$:    BBC     #2, SETFILE$FLAGS+1, 32$              ; 1327
            FD78  CD        24   AB B0 00327         MOVW    GBUF_VALUE, RECATTR+20                ; 1330
                  58             01 D0 0032D         MOVL    #1, STATUS                            ; 1331
                  19             58 E9 00330 32$:    BLBC    STATUS, 33$                           ; 1336
                          9A AD46 7F 00333           PUSHAQ  ATR+2[PTR]                            ; 1339
                  9E             04 B0 00337         MOVW    #4, @(SP)+                            ;
                          98 AD46 7F 0033A           PUSHAQ  ATR[PTR]                              ; 1340
                  9E             20 B0 0033E         MOVW    #32, @(SP)+                           ;
                          9C AD46 7F 00341           PUSHAQ  ATR+4[PTR]                            ; 1341
                  9E       FD64  CD 9E 00345         MOVAB   HEADER+20, @(SP)+                     ;
                                 56 D6 0034A         INCL    PTR                                   ; 1342
                  20        01   AB E9 0034C 33$:    BLBC    SETFILE$FLAGS+1, 34$                  ; 1348
FD76  CD          18  AB         08 28 00350         MOVC3   #8, EXP_VALUE, HEADER+38              ; 1351
                          9A AD46 7F 00357           PUSHAQ  ATR+2[PTR]                            ; 1352
                  9E             13 B0 0035B         MOVW    #19, @(SP)+                           ;
                          98 AD46 7F 0035E           PUSHAQ  ATR[PTR]                              ; 1353
                  9E             08 B0 00362         MOVW    #8, @(SP)+                            ;
                          9C AD46 7F 00365           PUSHAQ  ATR+4[PTR]                            ; 1354
                  9E       FD76  CD 9E 00369         MOVAB   HEADER+38, @(SP)+                     ;
```

```
                                              56  D6 0036E       INCL    PTR                                   ; 1355
      03         01    AB                     06  E0 00370 34$:  BBS     #6, SETFILE$FLAGS+1, 35$              ; 1361
                                            010A  31 00375       BRW     43$                                   ;
                                      01      AB  95 00378 35$:  TSTB    SETFILE$FLAGS+1                       ; 1371
                                              03  19 0037B       BLSS    37$                                   ;
                                            00E3  31 0037D 36$:  BRW     42$                                   ;
             00000000'  EF            2A      A7  B1 00380 37$:  CMPW    42(R7), OLD_DID_NUM                   ; 1374
                                              14  12 00388       BNEQ    38$                                   ;
             00000000'  EF            2C      A7  B1 0038A       CMPW    44(R7), OLD_DID_SEQ                   ; 1375
                                              0A  12 00392       BNEQ    38$                                   ;
             00000000'  EF            2E      A7  B1 00394       CMPW    46(R7), OLD_DID_RVN                   ; 1376
                                              DF  13 0039C       BEQL    36$                                   ;
                  0320  CE            39      A7  9B 0039E 38$:  MOVZBW  57(R7), TEMP_DESC                     ; 1385
                  FCDC  CD            44      A7  D0 003A4       MOVL    68(R7), TEMP_DESC+4                   ; 1386
                                              7E  7C 003AA       CLRQ    -(SP)                                 ; 1388
                                      10      AE  9F 003AC       PUSHAB  TEMP_CHAN                             ;
                                    FCD8      CD  9F 003AF       PUSHAB  TEMP_DESC                             ;
             00000000G  00                    04  FB 003B3       CALLS   #4, SYS$ASSIGN                        ;
                        58                    50  D0 003BA       MOVL    R0, STATUS                            ;
                        1A                    58  E8 003BD       BLBS    STATUS, 39$                           ;
                                              59  DD 003C0       PUSHL   R9                                    ; 1391
             00000000G  00                    01  FB 003C2       CALLS   #1, SYS$DASSGN                        ;
                                              58  DD 003C9       PUSHL   STATUS                                ; 1392
                                    018C      CB  9F 003CB       PUSHAB  CONF_DESC                             ;
                                              01  DD 003CF       PUSHL   #1                                    ;
                               00000000G      8F  DD 003D1       PUSHL   #SET$_OPENDIR                         ;
                                            FC81  31 003D7       BRW     1$                                    ;
   20            00                   6E      00  2C 003DA 39$:  MOVC5   #0, (SP), #0, #32, TEMP_FIB           ; 1396
                                    0300      CE     003DF                                                     ;
                  0300  CE          0401      8F  3C 003E2       MOVZWL  #1025, TEMP_FIB                       ; 1398
                  0304  CE            2A      A7  D0 003E9       MOVL    42(R7), TEMP_FIB+4                    ; 1399
                  0308  CE            2E      A7  B0 003EF       MOVW    46(R7), TEMP_FIB+8                    ; 1401
                  FCE0  CD      00150004      8F  D0 003F5       MOVL    #1376260, TEMP_ATR                    ; 1404
                  FCE4  CD            28      AB  9E 003FE       MOVAB   UIC_VALUE, TEMP_ATR+4                 ; 1405
                                    FCE8      CD  D4 00404       CLRL    TEMP_ATR+8                            ; 1406
                  0320  CE                    20  B0 00408       MOVW    #32, TEMP_DESC                        ; 1408
                  FCDC  CD          0300      CE  9E 0040D       MOVAB   TEMP_FIB, TEMP_DESC+4                 ; 1409
                                              7E  D4 00414       CLRL    -(SP)                                 ; 1415
                                    FCE0      CD  9F 00416       PUSHAB  TEMP_ATR                              ;
                                              7E  7C 0041A       CLRQ    -(SP)                                 ;
                                              7E  D4 0041C       CLRL    -(SP)                                 ;
                                    FCD8      CD  9F 0041E       PUSHAB  TEMP_DESC                             ;
                                              7E  7C 00422       CLRQ    -(SP)                                 ;
                                    FCF0      CD  9F 00424       PUSHAB  IOSB                                  ;
                                              32  DD 00428       PUSHL   #50                                   ;
                                      30      AE  DD 0042A       PUSHL   TEMP_CHAN                             ;
                                              7E  D4 0042D       CLRL    -(SP)                                 ;
             00000000G  00                    0C  FB 0042F       CALLS   #12, SYS$QIOW                         ;
                        58                    50  D0 00436       MOVL    R0, STATUS                            ;
                        08                    58  E9 00439       BLBC    STATUS, 40$                           ; 1416
                        58          FCF0      CD  3C 0043C       MOVZWL  IOSB, STATUS                          ;
                        15                    58  E8 00441       BLBS    STATUS, 41$                           ; 1417
                                              58  DD 00444 40$:  PUSHL   STATUS                                ; 1418
                                    018C      CB  9F 00446       PUSHAB  CONF_DESC                             ;
                                              01  DD 0044A       PUSHL   #1                                    ;
                               00000000G      8F  DD 0044C       PUSHL   #SET$_OPENDIR                         ;
             00000000G  00                    04  FB 00452       CALLS   #4, LIB$STOP                          ;
```

```
                                    08   AE   DD 00459 41$:    PUSHL   TEMP_CHAN                          ; 1419
                 00000000G  00           01   FB 0045C         CALLS   #1, SYS$DASSGN                     ;
                     FD8C   CD      28   AB   D0 00463 42$:    MOVL    UIC_VALUE, HEADER+60               ; 1423
                                 9A AD46  7F 00469             PUSHAQ  ATR+2[PTR]                         ; 1424
                            9E           15   B0 0046D         MOVW    #21, @(SP)+                        ;
                                 98 AD46  7F 00470             PUSHAQ  ATR[PTR]                           ; 1425
                            9E           04   B0 00474         MOVW    #4, @(SP)+                         ;
                                 9C AD46  7F 00477             PUSHAQ  ATR+4[PTR]                         ; 1426
                            9E    FD8C   CD   9E 0047B         MOVAB   HEADER+60, @(SP)+                  ;
                                         56   D6 00480         INCL    PTR                                ; 1427
              47       02   AB           01   E1 00482 43$:    BBC     #1, SETFILE$FLAGS+2, 45$           ; 1435
    02    FD64   CD         04           04   ED 00487         CMPZV   #4, #4, RECATTR, #2                ; 1437
                                         18   12 0048E         BNEQ    44$                                ;
                           00771302      8F   DD 00490         PUSHL   #7803650                           ; 1439
                               018C      CB   9F 00496         PUSHAB  CONF_DESC                          ;
                                         01   DD 0049A         PUSHL   #1                                 ;
                          00000000G      8F   DD 0049C         PUSHL   #SET$_WRITEERR                     ;
                 00000000G  00           04   FB 004A2         CALLS   #4, LIB$SIGNAL                     ;
                                         23   11 004A9         BRB     45$                                ;
                     FF67   CD           01   88 004AB 44$:    BISB2   #1, FIB+23                         ; 1442
                            50    FD6C   CD   3C 004B0         MOVZWL  RECATTR+8, R0                      ; 1443
              50            50           10   78 004B5         ASHL    #16, R0, R0                        ;
                            51    FD6E   CD   3C 004B9         MOVZWL  RECATTR+10, R1                     ; 1444
          FF6C   CD         50           51   C1 004BE         ADDL3   R1, R0, FIB+28                     ;
                                  FD70   CD   B5 004C4         TSTW    RECATTR+12                         ; 1445
                                         04   13 004C8         BEQL    45$                                ;
                                  FF6C   CD   D6 004CA         INCL    FIB+28                             ; 1446
              17       02   AB           03   E1 004CE 45$:    BBC     #3, SETFILE$FLAGS+2, 46$           ; 1452
                     FF5A   CD      2A   A7   D0 004D3         MOVL    42(R7), FIB+10                     ; 1455
                     FF5E   CD      2E   A7   B0 004D9         MOVW    46(R7), FIB+14                     ; 1457
                     FF65   CD           08   88 004DF         BISB2   #8, FIB+21                         ; 1459
                     FF7C   CD      34   AB   B0 004E4         MOVW    VRSN_VALUE, FIB+44                 ; 1461
                                         58   D4 004EA 46$:    CLRL    STATUS                             ; 1468
              1C       01   AB           03   E1 004EC         BBC     #3, SETFILE$FLAGS+1, 47$           ; 1470
                            01    FD57   CD   91 004F1         CMPB    HEADER+7, #1                       ; 1473
                                         18   12 004F6         BNEQ    48$                                ;
                          00000000'      EF   9F 004F8         PUSHAB  P.ADU                              ; 1476
                                         01   DD 004FE         PUSHL   #1                                 ; 1474
                          00000000G      8F   DD 00500         PUSHL   #SET$_NOTODS2                      ;
                 00000000G  00           03   FB 00506         CALLS   #3, LIB$SIGNAL                     ;
                                       009F   31 0050D 47$:    BRW     57$                                ;
              0A       09   AB           01   E0 00510 48$:    BBS     #1, SETFILE$JFLAGS+1, 49$          ; 1483
              05       09   AB           03   E0 00515         BBS     #3, SETFILE$JFLAGS+1, 49$          ;
                                    08   AB   95 0051A         TSTB    SETFILE$JFLAGS                     ; 1484
                                         05   18 0051D         BGEQ    50$                                ;
                     FD9B   CD           23   8A 0051F 49$:    BICB2   #35, HEADER+72                     ; 1489
                            0F      09   AB   E9 00524 50$:    BLBC    SETFILE$JFLAGS+1, 51$              ; 1492
         50       08   AB   01           07   EF 00528         EXTZV   #7, #1, SETFILE$JFLAGS, R0         ; 1496
  FD98   CD       01        01           50   F0 0052E         INSV    R0, #1, #1, HEADER+72              ;
                                         26   11 00535         BRB     53$                                ; 1497
              0F       09   AB           02   E1 00537 51$:    BBC     #2, SETFILE$JFLAGS+1, 52$          ; 1500
         50       09   AB   01           01   EF 0053C         EXTZV   #1, #1, SETFILE$JFLAGS+1, R0       ; 1504
  FD98   CD       01        00           50   F0 00542         INSV    R0, #0, #1, HEADER+72              ;
                                         12   11 00549         BRB     53$                                ; 1505
              14       09   AB           04   E1 0054B 52$:    BBC     #4, SETFILE$JFLAGS+1, 54$          ; 1508
         50       09   AB   01           03   EF 00550         EXTZV   #3, #1, SETFILE$JFLAGS+1, R0       ; 1512
  FD98   CD       01        05           50   F0 00556         INSV    R0, #5, #1, HEADER+72              ;
```

```
                        0C  AB              02  88 00055D 53$:    BISB2   #2, SETFILE$MFLAGS                       ; 1513
                        58                  01  D0 005561         MOVL    #1, STATUS                              ; 1514
                14      08  AB              02  E1 005564 54$:    BBC     #2, SETFILE$JFLAGS, 55$                 ; 1517
        50      08  AB                      01  EF 005569         EXTZV   #1, #1, SETFILE$JFLAGS, R0              ; 1521
FD98    CD          01                      03  F0 00056F         INSV    R0, #3, #1, HEADER+72
                        0C  AB              02  88 005576         BISB2   #2, SETFILE$MFLAGS                      ; 1522
                        58                  01  D0 00057A         MOVL    #1, STATUS                              ; 1523
                14      08  AB              04  E1 00057D 55$:    BBC     #4, SETFILE$JFLAGS, 56$                 ; 1525
        50      08  AB                      01  EF 005582         EXTZV   #3, #1, SETFILE$JFLAGS, R0              ; 1529
FD98    CD          01                      04  F0 005588         INSV    R0, #4, #1, HEADER+72
                        0C  AB              02  88 00058F         BISB2   #2, SETFILE$MFLAGS                      ; 1530
                        58                  01  D0 005593         MOVL    #1, STATUS                              ; 1531
                14      08  AB              06  E1 005596 56$:    BBC     #6, SETFILE$JFLAGS, 57$                 ; 1533
        50      08  AB                      01  EF 00559B         EXTZV   #5, #1, SETFILE$JFLAGS, R0              ; 1537
FD98    CD          01                      02  F0 005A1         INSV    R0, #2, #1, HEADER+72
                        0C  AB              02  88 005A8         BISB2   #2, SETFILE$MFLAGS                      ; 1538
                        58                  01  D0 005AC         MOVL    #1, STATUS                              ; 1539
                        01                  58  D1 005AF 57$:    CMPL    STATUS, #1                              ; 1548
                                            19  12 005B2         BNEQ    58$
                                    9A AD46 7F  005B4             PUSHAQ  ATR+2[PTR]                              ; 1551
                            9E          1D  B0 005B8             MOVW    #29, @(SP)+
                                    98 AD46 7F  005BB             PUSHAQ  ATR[PTR]                                ; 1552
                            9E          02  B0 005BF             MOVW    #2, @(SP)+
                                    9C AD46 7F  005C2             PUSHAQ  ATR+4[PTR]                              ; 1553
                            9E      FD98    CD  9E 005C6         MOVAB   HEADER+72, @(SP)+
                                            56  D6 005CB         INCL    PTR                                     ; 1554
                03          0C  AB              01  E0 005CD 58$:    BBS     #1, SETFILE$MFLAGS, 60$             ; 1562
                                        00BD    31  005D2 59$:    BRW     64$
                F8          0C  AB              02  E0 005D5 60$:    BBS     #2, SETFILE$MFLAGS, 59$
    00000C00'       EF  0E000820            8F  D0 005DA         MOVL    #234883104, RMSJNLID ACE                ; 1568
                        50      FD40        CD  9E 005E5         MOVAB   ITEM_LIST, $$ITMBLKPTR                  ; 1574
                        80  0022000C          8F  D0 005EA         MOVL    #2228236, ($$ITMBLKPTR)+
                        80      FCF8        CD  9E 005F1         MOVAB   LABEL_BUFFER, ($$ITMBLKPTR)+
                                            80  7C 005F6         CLRQ    ($$ITMBLKPTR)+
                                            7E  7C 005F8         CLRQ    -(SP)                                   ; 1577
                                            7E  D4 005FA         CLRL    -(SP)
                                    FCF0    CD  9F 005FC         PUSHAB  IOSB
                                    FD40    CD  9F 00600         PUSHAB  ITEM_LIST
                                            7E  D4 00604         CLRL    -(SP)
                                            59  DD 00606         PUSHL   R9
                                            01  DD 00608         PUSHL   #1
    00000000G       00                      08  FB 0060A         CALLS   #8, SYS$GETDVI
                        58                  50  D0 00611         MOVL    R0, STATUS
                        08                  58  E9 00614         BLBC    STATUS, 61$                             ; 1578
                                            01  DD 00617         PUSHL   #1                                      ; 1580
    00000000G       00                      01  FB 00619         CALLS   #1, SYS$WAITFR
                                            13  11 00620         BRB     62$
                                            7E  D4 00622 61$:    CLRL    -(SP)                                   ; 1582
                                            58  DD 00624         PUSHL   STATUS
                                            7E  D4 00626         CLRL    -(SP)
                            00771124        8F  DD 00628         PUSHL   #7803172
    00000000G       00                      04  FB 0062E         CALLS   #4, LIB$SIGNAL
                        16      FCF0        CD  E8 00635 62$:    BLBS    IOSB, 63$                               ; 1583
                                            7E  D4 0063A         CLRL    -(SP)                                   ; 1585
                        7E      FCF0        CD  3C 0063C         MOVZWL  IOSB, -(SP)
                                            7E  D4 00641         CLRL    -(SP)
                            00771124        8F  DD 00643         PUSHL   #7803172
```

```
                00000000G  00              04  FB 00649                CALLS   #4, LIB$SIGNAL                                                                  ;
00000000'  EF     FCF8  CD                0C  28 00650 63$:           MOVC3   #12, LABEL_BUFFER, RMSJNLID_ACE+4                                       ; 1587
                00000000'  EF          24   A7  D0 0065A              MOVL    36(R7), RMSJNLID_ACE+16                                                 ; 1589
                00000000'  EF          28   A7  B0 00662              MOVW    40(R7), RMSJNLID_ACE+20                                                 ; 1591
                        00000000'  EF      9F 0066A                    PUSHAB  RMSJNLID_ACE+24                                                         ; 1593
                00000000G  00              01  FB 00670                CALLS   #1, SYS$GETTIM                                                          ;
                        9A AD46            7F 00677                    PUSHAQ  ATR+2[PTR]                                                              ; 1595
                     9E                    1F  B0 0067B                MOVW    #31, @(SP)+                                                             ;
                        98 AD46            7F 0067E                    PUSHAQ  ATR[PTR]                                                                ; 1596
                     9E                    20  B0 00682                MOVW    #32, @(SP)+                                                             ;
                        9C AD46            7F 00685                    PUSHAQ  ATR+4[PTR]                                                              ; 1597
                     9E 00000000'  EF      9E 00689                    MOVAB   RMSJNLID_ACE, @(SP)+                                                    ;
                                           56  D6 00690                INCL    PTR                                                                     ; 1598
                3C        09     AB          05  E1 00692 64$:           BBC     #5, SETFILE$JFLAGS+1, 65$                                               ; 1610
                        9A AD46            7F 00697                    PUSHAQ  ATR+2[PTR]                                                              ; 1613
                     9E                    1F  B0 0069B                MOVW    #31, @(SP)+                                                             ;
                     50      0170     CB   3C 0069E                    MOVZWL  AI_JNL_DESC, R0                                                         ; 1614
                     50                    04  C0 006A3                ADDL2   #4, R0                                                                  ;
                        98 AD46            7F 006A6                    PUSHAQ  ATR[PTR]                                                                ;
                     9E                    50  B0 006AA                MOVW    R0, @(SP)+                                                              ;
                        9C AD46            7F 006AD                    PUSHAQ  ATR+4[PTR]                                                              ; 1615
                     9E      FD2C     CD   9E 006B1                    MOVAB   AI_JNL_ACE, @(SP)+                                                      ;
                                           56  D6 006B6                INCL    PTR                                                                     ; 1616
                FD2C    CD                 50  90 006B8                MOVB    R0, AI_JNL_ACE                                                          ; 1618
                FD2D    CD                 03  90 006BD                MOVB    #3, AI_JNL_ACE+1                                                        ; 1619
                FD2E    CD      0600     BF   B0 006C2                MOVW    #1536, AI_JNL_ACE+2                                                     ; 1620
FD30    CD      0174    DB      0170     CB   28 006C9                MOVC3   AI_JNL_DESC, @AI_JNL_DESC+4, AI_JNL_ACE+4                               ; 1624
                3C        09     AB          06  E1 006D3 65$:           BBC     #6, SETFILE$JFLAGS+1, 66$                                               ; 1631
                        9A AD46            7F 006D8                    PUSHAQ  ATR+2[PTR]                                                              ; 1634
                     9E                    1F  B0 006DC                MOVW    #31, @(SP)+                                                             ;
                     50      0178     CB   3C 006DF                    MOVZWL  AT_JNL_DESC, R0                                                         ; 1635
                     50                    04  C0 006E4                ADDL2   #4, R0                                                                  ;
                        98 AD46            7F 006E7                    PUSHAQ  ATR[PTR]                                                                ;
                     9E                    50  B0 006EB                MOVW    R0, @(SP)+                                                              ;
                        9C AD46            7F 006EE                    PUSHAQ  ATR+4[PTR]                                                              ; 1636
                     9E      FD18     CD   9E 006F2                    MOVAB   AT_JNL_ACE, @(SP)+                                                      ;
                                           56  D6 006F7                INCL    PTR                                                                     ; 1637
                FD18    CD                 50  90 006F9                MOVB    R0, AT_JNL_ACE                                                          ; 1639
                FD19    CD                 04  90 006FE                MOVB    #4, AT_JNL_ACE+1                                                        ; 1640
                FD1A    CD      0600     8F   B0 00703                MOVW    #1536, AT_JNL_ACE+2                                                     ; 1641
FD1C    CD      017C    DB      0178     CB   28 0070A                MOVC3   AT_JNL_DESC, @AT_JNL_DESC+4, AT_JNL_ACE+4                               ; 1645
                     09         AB          95  00714 66$:           TSTB    SETFILE$JFLAGS+1                                                        ; 1652
                                           3C  18 00717                BGEQ    67$                                                                     ;
                        9A AD46            7F 00719                    PUSHAQ  ATR+2[PTR]                                                              ; 1655
                     9E                    1F  B0 0071D                MOVW    #31, @(SP)+                                                             ;
                     50      0180     CB   3C 00720                    MOVZWL  BI_JNL_DESC, R0                                                         ; 1656
                     50                    04  C0 00725                ADDL2   #4, R0                                                                  ;
                        98 AD46            7F 00728                    PUSHAQ  ATR[PTR]                                                                ;
                     9E                    50  B0 0072C                MOVW    R0, @(SP)+                                                              ;
                        9C AD46            7F 0072F                    PUSHAQ  ATR+4[PTR]                                                              ; 1657
                     9E      FD04     CD   9E 00733                    MOVAB   BI_JNL_ACE, @(SP)+                                                      ;
                                           56  D6 00738                INCL    PTR                                                                     ; 1658
                FD04    CD                 50  90 0073A                MOVB    R0, BI_JNL_ACE                                                          ; 1660
                FD05    CD                 02  90 0073F                MOVB    #2, BI_JNL_ACE+1                                                        ; 1661
                FD06    CD      0600     8F   B0 00744                MOVW    #1536, BI_JNL_ACE+2                                                     ; 1662
FD08    CD      0184    DB      0180     CB   28 0074B                MOVC3   BI_JNL_DESC, @BI_JNL_DESC+4, BI_JNL_ACE+4                               ; 1666
                        80         AD       D4 00755 67$:           CLRL    FIB+48                                                                  ; 1674
```

```
                                     98 AD46   7F 00758          PUSHAQ  ATR[PTR]                              ; 1675
                                          9E   D4 0075C          CLRL    @(SP)+                                ;
                                          56   D5 0075E          TSTL    PTR                                   ; 1677
                                          0A   12 00760          BNEQ    68$                                   ;
                05          02  AB        01   E0 00762          BBS     #1, SETFILE$FLAGS+2, 68$              ; 1678
                73          02  AB        03   E1 00767          BBC     #3, SETFILE$FLAGS+2, 71$              ; 1679
       08       00              6E        00   2C 0076C  68$:    MOVC5   #0, (SP), #0, #8, RES_DESC            ; 1685
                                     FCE8 CD      00771                                                        ;
                      FCE8  CD       0200 8F   B0 00774          MOVW    #512, RES_DESC                        ; 1686
                      FCEC  CD       0130 CE   9E 0077B          MOVAB   RES_BUF, RES_DESC+4                   ; 1687
                                          7E   D4 00782          CLRL    -(SP)                                 ; 1694
                                       98 AD   9F 00784          PUSHAB  ATR                                   ;
                                     FCE8 CD   9F 00787          PUSHAB  RES_DESC                              ;
                                       18 AE   9F 0078B          PUSHAB  RES_LEN                               ;
                                          7E   D4 0078E          CLRL    -(SP)                                 ;
                                       90 AD   9F 00790          PUSHAB  DESC                                  ;
                                          7E   7C 00793          CLRQ    -(SP)                                 ;
                                     FCF0 CD   9F 00795          PUSHAB  IOSB                                  ;
                                          36   DD 00799          PUSHL   #54                                   ;
                                          59   DD 0079B          PUSHL   R9                                    ;
                                          7E   D4 0079D          CLRL    -(SP)                                 ;
                   00000000G  00          0C   FB 0079F          CALLS   #12, SYS$QIOW                         ;
                              58          50   D0 007A6          MOVL    R0, STATUS                            ;
                              08          58   E9 007A9          BLBC    STATUS, 69$                           ; 1695
                              58     FCF0 CD   3C 007AC          MOVZWL  IOSB, STATUS                          ;
                              13          58   E8 007B1          BLBS    STATUS, 70$                           ; 1696
                                     0500 8F   BB 007B4  69$:    PUSHR   #^M<R8,R10>                           ; 1697
                                00000000G 8F   DD 007B8          PUSHL   #SET$_WRITEERR                        ;
                   00000000V  EF          03   FB 007BE          CALLS   #3, FILE_ERROR                        ;
                                          18   11 007C5          BRB     71$                                   ;
                13          01  AB        04   E1 007C7  70$:    BBC     #4, SETFILE$FLAGS+1, 71$              ; 1699
                                     018C CB   9F 007CC          PUSHAB  CONF_DESC                             ; 1700
                                          01   DD 007D0          PUSHL   #1                                    ;
                                00000000G 8F   DD 007D2          PUSHL   #SET$_MODIFIED                        ;
                   00000000G  00          03   FB 007D8          CALLS   #3, LIB$SIGNAL                        ;
                                          7E   7C 007DF  71$:    CLRQ    -(SP)                                 ; 1710
                                          7E   7C 007E1          CLRQ    -(SP)                                 ;
                                          7E   D4 007E3          CLRL    -(SP)                                 ;
                                       90 AD   9F 007E5          PUSHAB  DESC                                  ;
                                          7E   7C 007E8          CLRQ    -(SP)                                 ;
                                     FCF0 CD   9F 007EA          PUSHAB  IOSB                                  ;
                                          34   DD 007EE          PUSHL   #52                                   ;
                                          59   DD 007F0          PUSHL   R9                                    ;
                                          7E   D4 007F2          CLRL    -(SP)                                 ;
                   00000000G  00          0C   FB 007F4          CALLS   #12, SYS$QIOW                         ;
                08          02  AB        05   E0 007FB  72$:    BBS     #5, SETFILE$FLAGS+2, 73$              ; 1717
                03          02  AB        04   E0 00800          BBS     #4, SETFILE$FLAGS+2, 73$              ;
                                        0385   31 00805          BRW     98$                                   ;
                      FF5A  CD         2A A7   D0 00808  73$:    MOVL    42(R7), FIB+10                        ; 1725
                      FF5E  CD         2E A7   B0 0080E          MOVW    46(R7), FIB+14                        ; 1727
                03          02  AB        05   E0 00814          BBS     #5, SETFILE$FLAGS+2, 74$              ; 1732
                                        00FC   31 00819          BRW     83$                                   ;
                              1A       34 A7   E8 0081C  74$:    BLBS    52(R7), 75$                           ; 1739
                15          34  A7        03   E0 00820          BBS     #3, 52(R7), 75$                       ; 1740
                               0077120A   8F   DD 00825          PUSHL   #7803402                              ; 1741
                                     018C CB   9F 0082B          PUSHAB  CONF_DESC                             ;
                                          01   DD 0082F          PUSHL   #1                                    ;
```

```
                                00000000G  8F   DD 00831          PUSHL   #SET$_REMERR                                    ;
                                     034C  31   00837          BRW     97$                                             ;
                             5D      03    AB   E8 0083A 75$:    BLBS    SETFILE$FLAGS+3, 79$                            ; 1752
59                           6B            03   E1 0083E          BBC     #3, SETFILE$FLAGS, 79$                          ; 1753
                             6E    018C    CB   9E 00842          MOVAB   CONF_DESC, (SP)                                 ; 1758
                                           5E   DD 00847          PUSHL   SP                                              ;
                                00000000'  EF   9F 00849          PUSHAB  P.ADW                                           ; 1757
                  00000000G  00            02   FB 0084F          CALLS   #2, LIB$CONFIRM_ACT                             ;
                             50            50   D0 00856          MOVL    R0, STATUS                                      ;
                             3F            58   E8 00859          BLBS    STATUS, 79$                                     ; 1759
                  00000000G  8F            58   D1 0085C          CMPL    STATUS, #LIB$_QUIPRO                            ; 1762
                                           03   12 00863          BNEQ    76$                                             ;
                                        0285    31 00865          BRW     89$                                             ;
                  00000000G  8F            58   D1 00868 76$:     CMPL    STATUS, #LIB$_QUICONACT                         ; 1764
                                           09   12 0086F          BNEQ    77$                                             ;
                        03   AB            01   88 00871          BISB2   #1, SETFILE$FLAGS+3                             ; 1765
                             50            01   D0 00875          MOVL    #1, STATUS                                      ;
                                           1E   11 00878          BRB     78$                                             ;
                  00000000G  8F            58   D1 0087A 77$:     CMPL    STATUS, #LIB$_NEGANS                            ; 1766
                                           15   13 00881          BEQL    78$                                             ;
                                           58   DD 00883          PUSHL   STATUS                                          ; 1767
                                    018C   CB   9F 00885          PUSHAB  CONF_DESC                                       ;
                                           01   DD 00889          PUSHL   #1                                              ;
                                00000000G  8F   DD 0088B          PUSHL   #SET$_WRITEERR                                  ;
                  00000000G  00            04   FB 00891          CALLS   #4, LIB$SIGNAL                                  ;
                             7A            58   E9 00898 78$:     BLBC    STATUS, 82$                                     ; 1769
                                FF54        CD   D4 0089B 79$:     CLRL    FIB+4                                           ; 1775
                                FF58        CD   B4 0089F          CLRW    FIB+8                                           ; 1777
                             50      3B    A7   9A 008A3          MOVZBL  59(R7), R0                                      ; 1784
                             51      3C    A7   9A 008A7          MOVZBL  60(R7), R1                                      ;
                             50            51   C0 008AB          ADDL2   R1, R0                                          ;
                             52      3D    A7   9A 008AE          MOVZBL  61(R7), R2                                      ; 1785
                0138  CB     50            52   C1 008B2          ADDL3   R2, R0, FILE_NAME                               ;
                        013C CB      4C    A7   D0 008B8          MOVL    76(R7), FILE_NAME+4                             ; 1786
                                           7E   7C 008BE          CLRQ    -(SP)                                           ; 1795
                                           7E   7C 008C0          CLRQ    -(SP)                                           ;
                                    0138   CB   9F 008C2          PUSHAB  FILE_NAME                                       ;
                                      90   AD   9F 008C6          PUSHAB  DESC                                            ;
                                           7E   7C 008C9          CLRQ    -(SP)                                           ;
                                    FCF0   CD   9F 008CB          PUSHAB  IOSB                                            ;
                                           35   DD 008CF          PUSHL   #53                                             ;
                             7E      2C    AE   3C 008D1          MOVZWL  CHANNEL, -(SP)                                  ;
                                           7E   D4 008D5          CLRL    -(SP)                                           ;
                  00000000G  00            0C   FB 008D7          CALLS   #12, SYS$QIOW                                   ;
                             58            50   D0 008DE          MOVL    R0, STATUS                                      ;
                             08            58   E9 008E1          BLBC    STATUS, 80$                                     ; 1796
                             58      FCF0  CD   3C 008E4          MOVZWL  IOSB, STATUS                                    ;
                             11            58   E8 008E9          BLBS    STATUS, 81$                                     ; 1797
                                           58   DD 008EC 80$:     PUSHL   STATUS                                          ; 1801
                                    0138   CB   9F 008EE          PUSHAB  FILE_NAME                                       ; 1798
                                           01   DD 008F2          PUSHL   #1                                              ;
                                00000000G  8F   DD 008F4          PUSHL   #SET$_WRITEERR                                  ;
                                     0289  31   008FA          BRW     97$                                             ;
                13      01    AB            04   E1 008FD 81$:     BBC     #4, SETFILE$FLAGS+1, 82$                        ; 1803
                                    018C   CB   9F 00902          PUSHAB  CONF_DESC                                       ; 1804
                                           01   DD 00906          PUSHL   #1                                              ;
                                00000000G  8F   DD 00908          PUSHL   #SET$_REMOVED                                   ;
```

```
                         00000000G  00               03  FB 0090E          CALLS    #3, LIB$SIGNAL                              ;
                                                   0275  31 00915 82$:     BRW      98$                                         ; 1750
 0050   8F           00              6E              00  2C 00918 83$:     MOVC5    #0, (SP), #0, #80, $RMS_PTR                 ; 1827
                                            02E0     CE     0091F                                                               ;
                          02E0  CE          5003     8F  B0 00922          MOVW     #20483, $RMS_PTR                            ;
                          02F6  CE                   02  90 00929          MOVB     #2, $RMS_PTR+22                             ;
                          02FF  CE                   02  90 0092E          MOVB     #2, $RMS_PTR+31                             ;
                          0308  CE          0280     CE  9E 00933          MOVAB    NEW_NAM, $RMS_PTR+40                        ;
                          030C  CE          013C     CB  D0 0093A          MOVL     FILE_NAME+4, $RMS_PTR+44                    ;
                          0314  CE          0138     CB  90 00941          MOVB     FILE_NAME, $RMS_PTR+52                      ;
 0060   8F           00              6E              00  2C 00948          MOVC5    #0, (SP), #0, #96, $RMS_PTR                 ; 1831
                                            0280     CE     0094F                                                               ;
                          0280  CE          6002     8F  B0 00952          MOVW     #24578, $RMS_PTR                            ;
                          028A  CE                   01  8E 00959          MNEGB    #1, $RMS_PTR+10                             ;
                          028C  CE          0118     CE  9E 0095E          MOVAB    NEW_NAM_EXP, $RMS_PTR+12                    ;
                          0290  CE                   57  D0 00965          MOVL     R7, $RMS_PTR+16                             ;
                     10     34  A7                   03  E1 0096A          BBC      #3, 52(R7), 84$                             ; 1836
                          0310  CE 00000000'         EF  9E 0096F          MOVAB    P.ADY, NEW_FAB+48                           ; 1839
                          0315  CE                   02  90 00978          MOVB     #2, NEW_FAB+53                              ; 1840
                                                     08  11 0097D          BRB      85$                                         ; 1836
                                            0310     CE  D4 0097F 84$:     CLRL     NEW_FAB+48                                  ; 1844
                                            0315     CE  94 00983          CLRB     NEW_FAB+53                                  ; 1845
                                            02E0     CE  9F 00987 85$:     PUSHAB   NEW_FAB                                     ; 1851
                         00000000G  00               01  FB 0098B          CALLS    #1, SYS$PARSE                               ;
                                    58               50  D0 00992          MOVL     R0, STATUS                                  ;
              0220  CE    0280  CE          0060     8F  28 00995          MOVC3    #96, NEW_NAM, NEW_NAM2                      ; 1852
                                            02E0     CE  9F 0099F          PUSHAB   NEW_FAB                                     ; 1853
                          F62C  CF                   01  FB 009A3          CALLS    #1, PARSE_NULL_STRING                       ;
                                1C                   58  E8 009A8          BLBS     STATUS, 86$                                 ; 1854
                                                     58  DD 009AB          PUSHL    STATUS                                      ; 1861
                                            0138     CB  9F 009AD          PUSHAB   FILE_NAME                                   ; 1857
                                            018C     CB  9F 009B1          PUSHAB   CONF_DESC                                   ;
                                                     02  DD 009B5          PUSHL    #2                                          ;
                                       00000000G     8F  DD 009B7          PUSHL    #SET$_ENTERR                                ;
                         00000000G  00               05  FB 009BD          CALLS    #5, LIB$STOP                                ;
                                                   0239  31 009C4          BRW      102$                                        ; 1862
 0050   8F           00              6E              00  2C 009C7 86$:     MOVC5    #0, (SP), #0, #80, $RMS_PTR                 ; 1872
                                            02E0     CE     009CE                                                               ;
                          02E0  CE          5003     8F  B0 009D1          MOVW     #20483, $RMS_PTR                            ;
                          02E4  CE      20000000     8F  D0 009D8          MOVL     #536870912, $RMS_PTR+4                      ;
                          02F6  CE                   02  90 009E1          MOVB     #2, $RMS_PTR+22                             ;
                          02FF  CE                   02  90 009E6          MOVB     #2, $RMS_PTR+31                             ;
                          0308  CE          0280     CE  9E 009EB          MOVAB    NEW_NAM, $RMS_PTR+40                        ;
                          030C  CE          022C     CE  D0 009F2          MOVL     NEW_NAM2+12, $RMS_PTR+44                    ;
                          0314  CE          022B     CE  90 009F9          MOVB     NEW_NAM2+11, $RMS_PTR+52                    ;
 0060   8F           00              6E              00  2C 00A00          MOVC5    #0, (SP), #0, #96, $RMS_PTR                 ; 1876
                                            0280     CE     00A07                                                               ;
                          0280  CE          6002     8F  B0 00A0A          MOVW     #24578, $RMS_PTR                            ;
                          028A  CE                   01  8E 00A11          MNEGB    #1, $RMS_PTR+10                             ;
                          028C  CE            18     AE  9E 00A16          MOVAB    NEW_NAM_EXP2, $RMS_PTR+12                   ;
                          0290  CE                   57  D0 00A1C          MOVL     R7, $RMS_PTR+16                             ;
                                            02E0     CE  9F 00A21          PUSHAB   NEW_FAB                                     ; 1878
                         00000000G  00               01  FB 00A25          CALLS    #1, SYS$PARSE                               ;
                                    58               50  D0 00A2C          MOVL     R0, STATUS                                  ;
              0220  CE    0280  CE          0060     8F  28 00A2F          MOVC3    #96, NEW_NAM, NEW_NAM2                      ; 1879
                                            02E0     CE  9F 00A39          PUSHAB   NEW_FAB                                     ; 1880
                          F592  CF                   01  FB 00A3D          CALLS    #1, PARSE_NULL_STRING                       ;
```

```
                        1C              58  E8  00A42          BLBS    STATUS, 87$                                     ;1881
                                        58  DD  00A45          PUSHL   STATUS                                          ;1888
                                0138    CB  9F  00A47          PUSHAB  FILE_NAME                                       ;1884
                                018C    CB  9F  00A4B          PUSHAB  CONF_DESC                                       ;
                                        02  DD  00A4F          PUSHL   #2                                              ;
                            00000000G   8F  DD  00A51          PUSHL   #SET$_ENTERR                                    ;
                  00000000G 00                  05  FB  00A57          CALLS   #5, LIB$SIGNAL                          ;
                                              019F  31  00A5E          BRW     102$                                    ;1889
                      FCE8  CD      022B      CE  9A  00A61  87$:     MOVZBL  NEW_NAM2+11, NEW_NAME                   ;1895
                      FCEC  CD      022C      CE  D0  00A68          MOVL    NEW_NAM2+12, NEW_NAME+4                 ;1896
                            50      025B      CE  9A  00A6F          MOVZBL  NEW_NAM2+59, R0                         ;1902
                            51      025C      CE  9A  00A74          MOVZBL  NEW_NAM2+60, R1                         ;
                            50          51    C0  00A79          ADDL2   R1, R0                                  ;
                            52      025D      CE  9A  00A7C          MOVZBL  NEW_NAM2+61, R2                         ;1903
              0218  CE      50          52    A1  00A81          ADDW3   R2, R0, NEW_DESC                        ;
                      021C  CE      026C      CE  D0  00A87          MOVL    NEW_NAM2+76, NEW_DESC+4                 ;1904
                      FF5A  CD      024A      CE  D0  00A8E          MOVL    NEW_NAM2+42, FIB+10                     ;1909
                      FF5E  CD      024E      CE  B0  00A95          MOVW    NEW_NAM2+46, FIB+14                     ;1911
                            51          14    A7  9A  00A9C          MOVZBL  20(R7), R1                              ;1918
                            50      0234      CE  9A  00AA0          MOVZBL  NEW_NAM2+20, R0                         ;1919
  50            00          15    A7          51  2D  00AA5          CMPC5   R1, 21(R7), #0, R0, NEW_NAM2+21         ;1918
                                0235    CE              00AAB                                                          ;
                                              09  13  00AAE          BEQL    88$                                     ;
                                    000184C4  8F  DD  00AB0          PUSHL   #99524                                  ;1920
                                              009F  31  00AB6          BRW     95$                                     ;
                      6A    03    AB              01  E0  00AB9  88$:     BBS     #1, SETFILE$FLAGS+3, 93$        ;1932
                      66          6B              03  E1  00ABE          BBC     #3, SETFILE$FLAGS, 93$          ;1933
                          10      AE      018C    CB  9E  00AC2          MOVAB   CONF_DESC, ARGLIST              ;1939
                          14      AE      FCE8    CD  9E  00AC8          MOVAB   NEW_NAME, ARGLIST+4             ;1940
                                          10      AE  9F  00ACE          PUSHAB  ARGLIST                         ;1941
                                    00000000'     EF  9F  00AD1          PUSHAB  P.ADZ                           ;
                  00000000G 00                  02  FB  00AD7          CALLS   #2, LIB$CONFIRM_ACT             ;
                            58                  50  D0  00ADE          MOVL    R0, STATUS                      ;
                            44                  58  E8  00AE1          BLBS    STATUS, 93$                     ;1943
                  00000000G 8F                  58  D1  00AE4          CMPL    STATUS, #LIB$_QUIPRO            ;1946
                                                08  12  00AEB          BNEQ    90$                             ;
                        02  AB          40      8F  88  00AED  89$:     BISB2   #64, SETFILE$FLAGS+2            ;1947
                                              010B  31  00AF2          BRW     102$                            ;
                  00000000G 8F                  58  D1  00AF5  90$:     CMPL    STATUS, #LIB$_QUICONACT         ;1948
                                                09  12  00AFC          BNEQ    91$                             ;
                        03  AB                  02  88  00AFE          BISB2   #2, SETFILE$FLAGS+3             ;1949
                            58                  01  D0  00B02          MOVL    #1, STATUS                      ;
                                                1E  11  00B05          BRB     92$                             ;
                  00000000G 8F                  58  D1  00B07  91$:     CMPL    STATUS, #LIB$_NEGANS            ;1950
                                                15  13  00B0E          BEQL    92$                             ;
                                                58  DD  00B10          PUSHL   STATUS                          ;1951
                                    018C        CB  9F  00B12          PUSHAB  CONF_DESC                       ;
                                                01  DD  00B16          PUSHL   #1                              ;
                                00000000G       8F  DD  00B18          PUSHL   #SET$_WRITEERR                  ;
                  00000000G 00                  04  FB  00B1E          CALLS   #4, LIB$SIGNAL                  ;
                            65                  58  E9  00B25  92$:     BLBC    STATUS, 98$                     ;1953
                                                7E  7C  00B28  93$:     CLRQ    -(SP)                           ;1966
                                                7E  7C  00B2A          CLRQ    -(SP)                           ;
                                    0228        CE  9F  00B2C          PUSHAB  NEW_DESC                        ;
                                      90        AD  9F  00B30          PUSHAB  DESC                            ;
                                                7E  7C  00B33          CLRQ    -(SP)                           ;
                                    FCF0        CD  9F  00B35          PUSHAB  IOSB                            ;
```

```
                                    33  DD 00B39          PUSHL   #51                              ;
              7E          2C        AE  3C 00B3B          MOVZWL  CHANNEL, -(SP)                   ;
                                    7E  D4 00B3F          CLRL    -(SP)                            ;
      00000000G  00                 0C  FB 00B41          CALLS   #12, SYS$QIOW                    ;
                 58                 50  D0 00B48          MOVL    R0, STATUS                       ;
                 08                 58  E9 00B4B          BLBC    STATUS, 94$                      ; 1967
                 58       FCF0      CD  3C 00B4E          MOVZWL  IOSB, STATUS                     ;
                 1B                 58  E8 00B53          BLBS    STATUS, 96$                      ; 1968
                                    58  DD 00B56  94$:    PUSHL   STATUS                           ; 1973
                          FCEB      CD  9F 00B58  95$:    PUSHAB  NEW_NAME                         ; 1969
                          018C      CB  9F 00B5C          PUSHAB  CONF_DESC                        ;
                                    02  DD 00B60          PUSHL   #2                               ;
                     00000000G      8F  DD 00B62          PUSHL   #SET$_ENTERR                     ;
      00000000G  00                 05  FB 00B68          CALLS   #5, LIB$SIGNAL                   ;
                                    1C  11 00B6F          BRB     98$                              ;
   17       01   AB                 04  E1 00B71  96$:    BBC     #4, SETFILE$FLAGS+1, 98$         ; 1975
                          FCEB      CD  9F 00B76          PUSHAB  NEW_NAME                         ; 1976
                          018C      CB  9F 00B7A          PUSHAB  CONF_DESC                        ;
                                    02  DD 00B7E          PUSHL   #2                               ;
                     00000000G      8F  DD 00B80          PUSHL   #SET$_ENTERED                    ;
      00000000G  00                 04  FB 00B86  97$:    CALLS   #4, LIB$SIGNAL                   ;
   24       02   AB                 02  E1 00B8D  98$:    BBC     #2, SETFILE$FLAGS+2, 99$         ; 1983
                                    5A  DD 00B92          PUSHL   R10                              ; 1985
      00000000V  EF                 01  FB 00B94          CALLS   #1, UNLOCK_ACTION                ;
                 58                 50  D0 00B9B          MOVL    R0, STATUS                       ;
                 15                 58  E8 00B9E          BLBS    STATUS, 99$                      ;
                                    58  DD 00BA1          PUSHL   STATUS                           ; 1987
                          018C      CB  9F 00BA3          PUSHAB  CONF_DESC                        ;
                                    01  DD 00BA7          PUSHL   #1                               ;
                     00000000G      8F  DD 00BA9          PUSHL   #SET$_UNLOCKERR                  ;
      00000000G  00                 04  FB 00BAF          CALLS   #4, LIB$SIGNAL                   ;
                 24       02        AB  E9 00BB6  99$:    BLBC    SETFILE$FLAGS+2, 100$            ; 1992
                                    5A  DD 00BBA          PUSHL   R10                              ; 1994
      00000000V  EF                 01  FB 00BBC          CALLS   #1, SETPRO_ACTION                ;
                 58                 50  D0 00BC3          MOVL    R0, STATUS                       ;
                 15                 58  E8 00BC6          BLBS    STATUS, 100$                     ;
                                    58  DD 00BC9          PUSHL   STATUS                           ; 1996
                          018C      CB  9F 00BCB          PUSHAB  CONF_DESC                        ;
                                    01  DD 00BCF          PUSHL   #1                               ;
                     00000000G      8F  DD 00BD1          PUSHL   #SET$_PROERR                     ;
      00000000G  00                 04  FB 00BD7          CALLS   #4, LIB$SIGNAL                   ;
                 7E       04        AE  3C 00BDE  100$:   MOVZWL  CHANNEL, -(SP)                   ; 2002
      00000000G  00                 01  FB 00BE2          CALLS   #1, SYS$DASSGN                   ;
                 58                 50  D0 00BE9          MOVL    R0, STATUS                       ;
                 11                 58  E8 00BEC          BLBS    STATUS, 102$                     ;
                          0500      8F  BB 00BEF          PUSHR   #^M<R8,R10>                      ; 2003
                     00000000G      8F  DD 00BF3          PUSHL   #SET$_CLOSEERR                   ;
      00000000V  EF                 03  FB 00BF9  101$:   CALLS   #3, FILE_ERROR                   ;
                 50                 01  D0 00C00  102$:   MOVL    #1, R0                           ; 2005
                                        04 00C03          RET                                      ; 2006

; Routine Size:  3076 bytes,    Routine Base:  $CODE$ + 0693
```

```
;  2016     2007  1 GLOBAL ROUTINE file_error (status1,status2,fab) =
;  2017     2008  1 !++
;  2018     2009  1 !
;  2019     2010  1 ! This routine is called if an error occured while trying to access
;  2020     2011  1 ! a file.  The kind of error is signalled, along with the file name.
;  2021     2012  1 !
;  2022     2013  1 !--
;  2023     2014  2 BEGIN
;  2024     2015  2
;  2025     2016  2 MAP
;  2026     2017  2     fab : REF $BBLOCK;                  ! Define the fab
;  2027     2018  2
;  2028     2019  2 BIND
;  2029     2020  2     status = status2 : $BBLOCK,
;  2030     2021  2     nam = .fab[fab$l_nam] : $BBLOCK;    ! Define the name block
;  2031     2022  2
;  2032     2023  2
;  2033     2024  2 LOCAL
;  2034     2025  2     desc : VECTOR[2];                   ! A temporary descriptor
;  2035     2026  2
;  2036     2027  2 !
;  2037     2028  2 ! Check to see if there's a name in the resultant string field.
;  2038     2029  2 ! If there is, use it.
;  2039     2030  2 !
;  2040     2031  2 IF .nam[nam$b_rsl] NEQ 0
;  2041     2032  2 THEN
;  2042     2033  3     BEGIN
;  2043     2034  3     desc[0] = .nam[nam$b_rsl];
;  2044     2035  3     desc[1] = .nam[nam$l_rsa];
;  2045     2036  3     END
;  2046     2037  3
;  2047     2038  3 !
;  2048     2039  3 ! If no resultant name, try the expanded name
;  2049     2040  3 !
;  2050     2041  2 ELSE IF .nam[nam$b_esl] NEQ 0
;  2051     2042  2 THEN
;  2052     2043  3     BEGIN
;  2053     2044  3     desc[0] = .nam[nam$b_esl];
;  2054     2045  3     desc[1] = .nam[nam$l_esa];
;  2055     2046  3     END
;  2056     2047  3
;  2057     2048  3 !
;  2058     2049  3 ! If no expanded name, use the original name in the fab
;  2059     2050  3 !
;  2060     2051  2 ELSE
;  2061     2052  3     BEGIN
;  2062     2053  3     desc[0] = .fab[fab$b_fns];
;  2063     2054  3     desc[1] = .fab[fab$l_fna];
;  2064     2055  2     END;
;  2065     2056  2
;  2066     2057  2 !
;  2067     2058  2 ! Signal the error
;  2068     2059  2 !
;  2069     2060  2 SIGNAL(.status1,                        ! Report error
;  2070     2061  2         1,                              ! One FAO argument
;  2071     2062  2         desc,                           ! Which is the file name
;  2072     2063  2         .status);                       ! Plus original error
```

```
; 2073          2064  2 RETURN true;
; 2074          2065  1 END;


                                        0000 00000          .ENTRY  FILE_ERROR, Save nothing                ; 2007
                                  5E    08  C2 00002          SUBL2   #8, SP
                                  51    0C  AC D0 00005          MOVL    FAB, R1                         ; 2021
                                  50    28  A1 D0 00009          MOVL    40(R1), R0
                                        03  A0 95 0000D          TSTB    3(R0)                           ; 2031
                                        0B  13 00010          BEQL    1$
                                  6E    03  A0 9A 00012          MOVZBL  3(R0), DESC                     ; 2034
                              04  AE    04  A0 D0 00016          MOVL    4(R0), DESC+4                   ; 2035
                                        19  11 0001B          BRB     3$                              ; 2031
                                        0B  A0 95 0001D 1$:      TSTB    11(R0)                          ; 2041
                                        0B  13 00020          BEQL    2$
                                  6E    0B  A0 9A 00022          MOVZBL  11(R0), DESC                    ; 2044
                              04  AE    0C  A0 D0 00026          MOVL    12(R0), DESC+4                  ; 2045
                                        09  11 0002B          BRB     3$                              ; 2041
                                  6E    34  A1 9A 0002D 2$:      MOVZBL  52(R1), DESC                    ; 2053
                              04  AE    2C  A1 D0 00031          MOVL    44(R1), DESC+4                  ; 2054
                                        08  AC DD 00036 3$:      PUSHL   STATUS                          ; 2063
                                        04  AE 9F 00039          PUSHAB  DESC                            ; 2060
                                        01  DD 0003C          PUSHL   #1
                                        04  AC DD 0003E          PUSHL   STATUS1
                          00000000G     00     04 FB 00041          CALLS   #4, LIB$SIGNAL
                                  50          01 D0 00048          MOVL    #1, R0                          ; 2064
                                              04 0004B          RET                                     ; 2065

; Routine Size:  76 bytes,    Routine Base:  $CODE$ + 1297

; 2075          2066  1
```

```
;  2077          2067  1 GLOBAL ROUTINE check_privilege : NOVALUE =
;  2078          2068  1 !++
;  2079          2069  1 !
;  2080          2070  1 ! This routine checks that the image has the correct privilege.
;  2081          2071  1 !
;  2082          2072  1 !---
;  2083          2073  2 BEGIN
;  2084          2074  2
;  2085          2075  2 LOCAL
;  2086          2076  2     status,
;  2087          2077  2     oldpriv : $BBLOCK[8];                ! Permanent privileges go here
;  2088          2078  2
;  2089          2079  2 OWN
;  2090          2080  2     newpriv : $BBLOCK[8]                          ! Mask to disable SYSPRV
;  2091          2081  2               PRESET([prv$v_sysprv]=true);
;  2092          2082  2
;  2093          2083  2 !
;  2094          2084  2 ! The image SET is installed with SYSPRV privilege, but we don't want the user
;  2095          2085  2 ! to have that much power unless s/he already has it.  So, first check to
;  2096          2086  2 ! see if the process has the privilege, and if not, then remove it for the
;  2097          2087  2 ! duration of this image.
;  2098          2088  2 !
;  2099        P 2089  2 IF NOT (status = $SETPRV(ENBFLG = 1,                 ! Enable
;  2100        P 2090  2                          PRVADR = 0,                 ! No new privileges
;  2101        P 2091  2                          PRMFLG = 1,                 ! Permanent privs
;  2102          2092  2                          PRVPRV = oldpriv))          ! Store current ones here
;  2103          2093  2 THEN SIGNAL_STOP(.status);
;  2104          2094  2 !
;  2105          2095  2 ! Check to see if privilege there.  If not, then remove it from current
;  2106          2096  2 ! privileges.
;  2107          2097  2 !
;  2108          2098  2 IF NOT .oldpriv[prv$v_sysprv]                     ! If SYSPRV not permanent
;  2109          2099  2 THEN
;  2110          2100  3     BEGIN
;  2111        P 2101  4     IF NOT (status = $SETPRV(ENBFLG = 0,          ! Disable
;  2112        P 2102  4                              PRVADR = newpriv,   ! this privilege
;  2113        P 2103  4                              PRMFLG = 0,         ! for the duration of this image
;  2114          2104  4                              PRVPRV = 0))
;  2115          2105  3     THEN SIGNAL_STOP(.status)
;  2116          2106  2     END;
;  2117          2107  2
;  2118          2108  2 RETURN;
;  2119          2109  1 END;


                                                .PSECT  $OWN$,NOEXE,2

                                    0002A       .BLKB   2
                                00# 0002C NEWPRIV:.BYTE   0[3]
                                10  0002F       .BYTE   16
                                    00030       .BLKB   4


                                                .PSECT  $CODE$,NOWRT,2
```

```
                                    001C 00000          .ENTRY  CHECK_PRIVILEGE, Save R2,R3,R4          ; 2067
                            54 00000000G   00    9E 00002       MOVAB   SYS$SETPRV, R4                  ;
                            53 00000000G   00    9E 00009       MOVAB   LIB$STOP, R3                    ;
                            5E             08    C2 00010       SUBL2   #8, SP                          ;
                                           5E    DD 00013       PUSHL   SP                              ; 2092
                                           01    DD 00015       PUSHL   #1                              ;
                            7E             01    7D 00017       MOVQ    #1, -(SP)                       ;
                            64             04    FB 0001A       CALLS   #4, SYS$SETPRV                  ;
                            52             50    D0 0001D       MOVL    R0, STATUS                      ;
                            05             52    E8 00020       BLBS    STATUS, 1$                      ;
                                           52    DD 00023       PUSHL   STATUS                          ; 2093
                            63             01    FB 00025       CALLS   #1, LIB$STOP                    ;
                18      03  AE             04    E0 00028 1$:   BBS     #4, OLDPRIV+3, 2$               ; 2098
                                           7E    7C 0002D       CLRQ    -(SP)                           ; 2104
                                00000000'  EF    9F 0002F       PUSHAB  NEWPRIV                         ;
                                           7E    D4 00035       CLRL    -(SP)                           ;
                            64             04    FB 00037       CALLS   #4, SYS$SETPRV                  ;
                            52             50    D0 0003A       MOVL    R0, STATUS                      ;
                            05             52    E8 0003D       BLBS    STATUS, 2$                      ;
                                           52    DD 00040       PUSHL   STATUS                          ; 2105
                            63             01    FB 00042       CALLS   #1, LIB$STOP                    ;
                                                  04 00045 2$:    RET                                     ; 2109

; Routine Size:  70 bytes,    Routine Base:  $CODE$ + 12E3
```

```
;  2121          2110  1 GLOBAL ROUTINE search_error (fab) =
;  2122          2111  1 !++
;  2123          2112  1 !
;  2124          2113  1 ! This routine is called when lib$file_scan detects an error while
;  2125          2114  1 ! searching for a file specified in the command line.
;  2126          2115  1 !
;  2127          2116  1 !--
;  2128          2117  2 BEGIN
;  2129          2118  2
;  2130          2119  2 MAP
;  2131          2120  2     fab : REF $BBLOCK;                  ! Define FAB format
;  2132          2121  2
;  2133          2122  2 BIND
;  2134          2123  2     nam = .fab[fab$l_nam] : $BBLOCK;    ! Define NAM block
;  2135          2124  2
;  2136          2125  2 LOCAL
;  2137          2126  2     desc : VECTOR[2];                   ! A temporary descriptor
;  2138          2127  2
;  2139          2128  2 !
;  2140          2129  2 ! Check to see if there's a name in the resultant string field.
;  2141          2130  2 ! If there is, use it.
;  2142          2131  2 !
;  2143          2132  2 IF .nam[nam$b_rsl] NEQ 0
;  2144          2133  2 THEN
;  2145          2134  3     BEGIN
;  2146          2135  3     desc[0] = .nam[nam$b_rsl];
;  2147          2136  3     desc[1] = .nam[nam$l_rsa];
;  2148          2137  3     END
;  2149          2138  2
;  2150          2139  2 !
;  2151          2140  2 ! If no resultant name, try the expanded name
;  2152          2141  2 !
;  2153          2142  2 ELSE IF .nam[nam$b_esl] NEQ 0
;  2154          2143  2 THEN
;  2155          2144  3     BEGIN
;  2156          2145  3     desc[0] = .nam[nam$b_esl];
;  2157          2146  3     desc[1] = .nam[nam$l_esa];
;  2158          2147  3     END
;  2159          2148  2
;  2160          2149  2 !
;  2161          2150  2 ! If no expanded name, use the original name in the fab
;  2162          2151  2 !
;  2163          2152  2 ELSE
;  2164          2153  3     BEGIN
;  2165          2154  3     desc[0] = .fab[fab$b_fns];
;  2166          2155  3     desc[1] = .fab[fab$l_fna];
;  2167          2156  2     END;
;  2168          2157  2
;  2169          2158  2 !
;  2170          2159  2 ! Signal the error
;  2171          2160  2 !
;  2172          2161  2 SIGNAL_STOP(set$_searchfail,
;  2173          2162  2                1,                     ! One FAO argument
;  2174          2163  2                desc,                  ! Which is the file name
;  2175          2164  2                .fab[fab$l_sts],       ! Show RMS error code
;  2176          2165  2                .fab[fab$l_stv]);      ! And secondary error code
;  2177          2166  2 RETURN true;
```

```
; 2178          2167  1 END;


                                        0000 00000          .ENTRY  SEARCH_ERROR, Save nothing                       ; 2110
                                    5E  08   C2 00002           SUBL2   #8, SP                                          ;
                                    51  04   AC D0 00005           MOVL    FAB, R1                                     ; 2123
                                    50  28   A1 D0 00009           MOVL    40(R1), R0                                  ;
                                        03   A0 95 0000D           TSTB    3(R0)                                       ; 2132
                                        08   13 00010           BEQL    1$                                          ;
                                    6E  03   A0 9A 00012           MOVZBL  3(R0), DESC                                 ; 2135
                                04  AE  04   A0 D0 00016           MOVL    4(R0), DESC+4                               ; 2136
                                        19   11 0001B           BRB     3$                                          ; 2132
                                        08   A0 95 0001D 1$:       TSTB    11(R0)                                      ; 2142
                                        0B   13 00020           BEQL    2$                                          ;
                                    6E  08   A0 9A 00022           MOVZBL  11(R0), DESC                                ; 2145
                                04  AE  0C   A0 D0 00026           MOVL    12(R0), DESC+4                              ; 2146
                                        09   11 0002B           BRB     3$                                          ; 2142
                                    6E  34   A1 9A 0002D 2$:       MOVZBL  52(R1), DESC                                ; 2154
                                04  AE  2C   A1 D0 00031           MOVL    44(R1), DESC+4                              ; 2155
                                    7E  08   A1 7D 00036 3$:       MOVQ    8(R1), -(SP)                                ; 2164
                                        08   AE 9F 0003A           PUSHAB  DESC                                        ; 2161
                                        01   DD 0003D           PUSHL   #1                                          ;
                                 0077123A   8F DD 0003F           PUSHL   #7803450                                    ;
                        00000000G   00      05 FB 00045           CALLS   #5, LIB$STOP                                ;
                                    50      01 D0 0004C           MOVL    #1, R0                                      ; 2166
                                            04 0004F           RET                                                     ; 2167

; Routine Size:  80 bytes,    Routine Base:  $CODE$ + 1329
```

```
;  2180          2168  1 ROUTINE unlock_action (fab)  =
;  2181          2169  1
;  2182          2170  1 !----
;  2183          2171  1 !
;  2184          2172  1 ! Functional description
;  2185          2173  1 !
;  2186          2174  1 !       This routine is called from SET ATTRIBUTES whenever
;  2187          2175  1 !       a successful file match for /LOCK occurs
;  2188          2176  1 !
;  2189          2177  1 ! Input parameters
;  2190          2178  1 !
;  2191          2179  1 !       fab = Address of block describing the file
;  2192          2180  1 !
;  2193          2181  1 ! Output parameters
;  2194          2182  1 !
;  2195          2183  1 !       None
;  2196          2184  1 !
;  2197          2185  1 !----
;  2198          2186  1
;  2199          2187  2 BEGIN
;  2200          2188  2
;  2201          2189  2 MAP fab: REF $BBLOCK;                    ! Define fab block format
;  2202          2190  2
;  2203          2191  2 LOCAL status;                            ! Receives status
;  2204          2192  2
;  2205          2193  2 !
;  2206          2194  2 ! If /CONFIRM was set by the user then interrogate him to see if
;  2207          2195  2 ! this file is to be unlocked
;  2208          2196  2 !
;  2209          2197  2 IF
;  2210          2198  3 BEGIN
;  2211          2199  3     IF .setfile$flags[qual_quit_unlock]
;  2212          2200  3     OR NOT .setfile$flags[qual_confirm]
;  2213          2201  3     THEN true
;  2214          2202  3     ELSE
;  2215          2203  4         BEGIN
;  2216          2204  4         status = lib$confirm_act(%ASCID 'Unlock file !AS? [N] : ',
;  2217          2205  4                                  %REF(conf_desc));
;  2218          2206  4         IF NOT .status
;  2219          2207  4         THEN
;  2220          2208  5             BEGIN
;  2221          2209  5             IF .status EQL lib$_quipro
;  2222          2210  6             THEN (setfile$flags[qual_quit] = 1; RETURN true)
;  2223          2211  5             ELSE IF .status EQL lib$_quiconact
;  2224          2212  6             THEN (setfile$flags[qual_quit_mod] = 1; status = 1)
;  2225          2213  5             ELSE IF .status NEQ lib$_negans
;  2226          2214  5             THEN SIGNAL(set$_writeerr, 1, conf_desc, .status);
;  2227          2215  4             END;
;  2228          2216  4         .status
;  2229          2217  4         END
;  2230          2218  3     END
;  2231          2219  2 THEN
;  2232          2220  3 BEGIN
;  2233          2221  3
;  2234          2222  3 !
;  2235          2223  3 ! Call LIB$UNLOCK_FILE to unlock the file
;  2236          2224  3 !
```

```
;  2237          2225  3
;  2238          2226  4      IF NOT (status = lib$unlock_file(conf_desc))        ! Call unlock with file name
;  2239          2227  3      THEN
;  2240          2228  3          RETURN(.status);
;  2241          2229  3
;  2242          2230  3  !
;  2243          2231  3  ! Check to see if unlock worked.  SS$_WASSET indicates the file
;  2244          2232  3  ! was unlocked.  SS$_WASCLR indicates the file was already unlocked
;  2245          2233  3  ! and no other error occurred
;  2246          2234  3  !
;  2247          2235  3
;  2248          2236  4      IF (.status EQL SS$_WASCLR)          ! If file not locked
;  2249          2237  3      THEN
;  2250          2238  3          SIGNAL(set$_notlocked,1,conf_desc)
;  2251          2239  3      ELSE                                             ! File was unlocked
;  2252          2240  3          IF .setfile$flags[qual_log]                  !if /LOG tell user
;  2253          2241  3          THEN SIGNAL(set$_unlocked,1,conf_desc);
;  2254          2242  2      END;
;  2255          2243  2      RETURN(true);                                    ! Both returns above
;  2256          2244  2                                                       !   are ok!
;  2257          2245  2
;  2258          2246  1  END;


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

53 41 21 20 65 6C 69 66 20 6B 63 6F 6C 6E 55  00458 P.AEC:  .ASCII  \Unlock file !AS? [N] : \<0>                    ;
                  00 20 3A 20 5D 4E 5B 20 3F  00467                                                               ;
                                 010E0017     00470 P.AEB:  .LONG   17694743                                      ;
                                 00000000'    00474         .ADDRESS P.AEC                                        ;


                                                        .PSECT  $CODE$,NOWRT,2

                                          001C 00000 UNLOCK_ACTION:
                                                        .WORD   Save R2,R3,R4                                   ; 2168
                                 54 00000000G   00 9E 00002     MOVAB   LIB$SIGNAL, R4                          ;
                                 53 00000000'   EF 9E 00009     MOVAB   SETFILE$FLAGS, R3                       ;
                                 5E             04 C2 00010     SUBL2   #4, SP                                  ;
5E          03                   A3             03 E0 00013     BBS     #3, SETFILE$FLAGS+3, 4$                 ; 2199
5A                               63             03 E1 00018     BBC     #3, SETFILE$FLAGS, 4$                   ; 2200
                                 6E       018C  C3 9E 0001C     MOVAB   CONF_DESC, (SP)                         ; 2205
                                 5E             DD 00021        PUSHL   SP                                      ;
                                 00000000'      EF 9F 00023     PUSHAB  P.AEB                                   ; 2204
                00000000G        00             02 FB 00029     CALLS   #2, LIB$CONFIRM_ACT                     ;
                                 52             50 D0 00030     MOVL    R0, STATUS                              ;
                                 40             52 E8 00033     BLBS    STATUS, 4$                              ; 2206
                00000000G        8F             52 D1 00036     CMPL    STATUS, #LIB$_QUIPRO                    ; 2209
                                                07 12 0003D     BNEQ    1$                                      ;
                      02         A3       40    8F 88 0003F     BISB2   #64, SETFILE$FLAGS+2                    ; 2210
                                                6C 11 00044     BRB     8$                                      ;
                00000000G        8F             52 D1 00046 1$: CMPL    STATUS, #LIB$_QUICONACT                 ; 2211
                                                0A 12 0004D     BNEQ    2$                                      ;
                      02         A3       80    8F 88 0004F     BISB2   #128, SETFILE$FLAGS+2                   ; 2212
                                 52             01 D0 00054     MOVL    #1, STATUS                              ;
```

```
                                   1A  11 00057       BRB     3$                               ;
          00000000G  8F            52  D1 00059 2$:   CMPL    STATUS, #LIB$_NEGANS             ; 2213
                                   11  13 00060       BEQL    3$                               ;
                                   52  DD 00062       PUSHL   STATUS                           ; 2214
                          018C     C3  9F 00064       PUSHAB  CONF_DESC                        ;
                                   01  DD 00068       PUSHL   #1                               ;
                     00000000G     8F  DD 0006A       PUSHL   #SET$_WRITEERR                   ;
                     64            04  FB 00070       CALLS   #4, LIB$SIGNAL                   ;
                     3C            52  E9 00073 3$:   BLBC    STATUS, 8$                       ; 2216
                          018C     C3  9F 00076 4$:   PUSHAB  CONF_DESC                        ; 2226
          00000000G  00            01  FB 0007A       CALLS   #1, LIB$UNLOCK_FILE              ;
                     52            50  D0 00081       MOVL    R0, STATUS                       ;
                     04            52  E8 00084       BLBS    STATUS, 5$                       ;
                     50            52  D0 00087       MOVL    STATUS, R0                       ; 2228
                                   04     0008A       RET                                      ;
                     01            52  D1 0008B 5$:   CMPL    STATUS, #1                       ; 2236
                                   0E  12 0008E       BNEQ    6$                               ;
                          018C     C3  9F 00090       PUSHAB  CONF_DESC                        ; 2238
                                   01  DD 00094       PUSHL   #1                               ;
                     00000000G     8F  DD 00096       PUSHL   #SET$_NOTLOCKED                  ;
                                   11  11 0009C       BRB     7$                               ;
      0F        01   A3            04  E1 0009E 6$:   BBC     #4, SETFILE$FLAGS+1, 8$          ; 2240
                          018C     C3  9F 000A3       PUSHAB  CONF_DESC                        ; 2241
                                   01  DD 000A7       PUSHL   #1                               ;
                     00000000G     8F  DD 000A9       PUSHL   #SET$_UNLOCKED                   ;
                     64            03  FB 000AF 7$:   CALLS   #3, LIB$SIGNAL                   ;
                     50            01  D0 000B2 8$:   MOVL    #1, R0                           ; 2243
                                   04     000B5       RET                                      ; 2246

; Routine Size:  182 bytes,    Routine Base:  $CODE$ + 1379
```

```
;  2260          2247  1 ROUTINE setpro_action (fab): =
;  2261          2248  1
;  2262          2249  1 !----
;  2263          2250  1 !
;  2264          2251  1 ! Functional description
;  2265          2252  1 !
;  2266          2253  1 !       This routine is called from SET_ATTRIBUTES whenever
;  2267          2254  1 !       the qualifier PROTECTION is found
;  2268          2255  1 !
;  2269          2256  1 ! Input parameters
;  2270          2257  1 !
;  2271          2258  1 !       fab = Address of block describing the file
;  2272          2259  1 !       fab$l_nam = pointer to name block
;  2273          2260  1 !
;  2274          2261  1 ! Output parameters
;  2275          2262  1 !
;  2276          2263  1 !       First error encountered, or TRUE is RETURNed
;  2277          2264  1 !
;  2278          2265  1 !----
;  2279          2266  1
;  2280          2267  2 BEGIN
;  2281          2268  2
;  2282          2269  2 MAP fab: REF $BBLOCK;                    ! Define fab block format
;  2283          2270  2
;  2284          2271  2 LOCAL
;  2285          2272  2     p_res_mask,                          ! Enable-mask parameter
;  2286          2273  2     p_res_prot,                          ! Value-mask parameter
;  2287          2274  2     final_prot: WORD,                    ! Receives final protection
;  2288          2275  2     desc:       VECTOR[2],               ! Temporary string descriptor
;  2289          2276  2     status;                              ! Receives status
;  2290          2277  2
;  2291          2278  2
;  2292          2279  2 !
;  2293          2280  2 ! If /CONFIRM was set by the user then interrogate him to see if
;  2294          2281  2 ! this file is to have its protection changed.
;  2295          2282  2 !
;  2296          2283  2 IF
;  2297          2284  3 BEGIN
;  2298          2285  3     IF .setfile$flags[qual_quit_protect]
;  2299          2286  3     OR NOT .setfile$flags[qual_confirm]
;  2300          2287  3     THEN true
;  2301          2288  3     ELSE
;  2302          2289  4         BEGIN
;  2303          2290  4         status = lib$confirm_act(%ASCID 'Change protection of file !AS? [N] : ',
;  2304          2291  4                                  %REF(conf_desc));
;  2305          2292  4         IF NOT .status
;  2306          2293  4         THEN
;  2307          2294  5             BEGIN
;  2308          2295  5             IF .status EQL lib$_quipro
;  2309          2296  6             THEN (setfile$flags[qual_quit] = 1; RETURN true)
;  2310          2297  5             ELSE IF .status EQL lib$_quiconact
;  2311          2298  6             THEN (setfile$flags[qual_quit_mod] = 1; status = 1)
;  2312          2299  5             ELSE IF .status NEQ lib$_negans
;  2313          2300  5             THEN SIGNAL(set$_writeerr, 1, conf_desc, .status);
;  2314          2301  4             END;
;  2315          2302  4         .status
;  2316          2303  4         END
```

```
;  2317          2304  3     END
;  2318          2305  2 THEN
;  2319          2306  2 BEGIN
;  2320          2307  2
;  2321          2308  2 ! Compute the parameters for lib$set_file_prot.  If not protection
;  2322          2309  2 ! value was specified set  enable-mask and value-mask to zero to
;  2323          2310  2 ! cause protection to be set to the process default.
;  2324          2311  2
;  2325          2312  2 p_res_mask = global_mask;
;  2326          2313  2 p_res_prot = global_prot;
;  2327          2314  2
;  2328          2315  2 IF .global_mask EQL 0                    ! If not protection values specified
;  2329          2316  2    THEN p_res_mask = p_res_prot = 0;     ! pass null parameters
;  2330          2317  2
;  2331          2318  2
;  2332          2319  2 !
;  2333          2320  2 ! Call lib$set_file_prot to set file protection
;  2334          2321  2 !
;  2335          2322  2
;  2336          2323  4 IF NOT (status = lib$set_file_prot (     ! Call library routine with
;  2337          2324  4                         conf_desc,      !  - file name
;  2338          2325  4                         .p_res_mask,    !  - result mask
;  2339          2326  4                         .p_res_prot,    !  - result protection
;  2340          2327  4                         final_prot))    !  - final protection returned
;  2341          2328  4                                         !    by lib$set_file_prot
;  2342          2329  3     THEN
;  2343          2330  4         BEGIN
;  2344          2331  4         SIGNAL  (                       ! Tell the user of error
;  2345          2332  4                         set$_pronotchg, !  - "Not changed" error message
;  2346          2333  4                         1,              !  - 1 FAO arguement
;  2347          2334  4                         conf_desc,      !  - descriptor of filename
;  2348          2335  4                         .status);       !  - original error
;  2349          2336  4         return (.status);               ! Return to the caller
;  2350          2337  3         END;
;  2351          2338  3
;  2352          2339  3
;  2353          2340  3 !
;  2354          2341  3 ! If /LOG was set then do it
;  2355          2342  3 !
;  2356          2343  3
;  2357          2344  4 IF (.setfile$flags[qual_log])           ! If logging requested
;  2358          2345  3 THEN prot_log_results (.fab..final_prot);       !  then call the logger
;  2359          2346  3
;  2360          2347  2 END;
;  2361          2348  2 RETURN (true);
;  2362          2349  2
;  2363          2350  1 END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

69  74  63  65  74  6F  72  70  20  65  67  6E  61  68  43  00478 P.AEE:  .ASCII  \Change protection of file !AS? [N] : \<0>  ;
3F  53  41  21  20  65  6C  69  66  20  66  6F  20  6E  6F  00487                                                            ;
                    00  20  3A  20  5D  4E  5B  20  00496                                                                  ;
                                        00  00  0049E               .ASCII  <0><0>                                         ;
                                010E0025    004A0 P.AED:  .LONG   17694757                                               ;
```

```
                        00000000' 004A4          .ADDRESS P.AEE                                          ;


                                                 .PSECT  $CODE$,NOWRT,2

                                  001C 00000 SETPRO_ACTION:
                                             .WORD   Save R2,R3,R4                                       ; 2247
                        54 00000000G   00  9E 00002          MOVAB   LIB$SIGNAL, R4                      ;
                        53 00000000'   EF  9E 00009          MOVAB   SETFILE$FLAGS, R3                   ;
                        5E             10  C2 00010          SUBL2   #16, SP                             ;
5E      03              A3             02  E0 00013          BBS     #2, SETFILE$FLAGS+3, 4$             ; 2285
5A                      63             03  E1 00018          BBC     #3, SETFILE$FLAGS, 4$               ; 2286
                        6E       018C  C3  9E 0001C          MOVAB   CONF_DESC, (SP)                     ; 2291
                                       5E  DD 00021          PUSHL   SP                                  ;
                        00000000'      EF  9F 00023          PUSHAB  P.AED                               ; 2290
          00000000G     00             02  FB 00029          CALLS   #2, LIB$CONFIRM_ACT                 ;
                        52             50  D0 00030          MOVL    R0, STATUS                          ;
                        40             52  E8 00033          BLBS    STATUS, 4$                          ; 2292
          00000000G     8F             52  D1 00036          CMPL    STATUS, #LIB$_QUIPRO                ; 2295
                                       07  12 0003D          BNEQ    1$                                  ;
               02       A3       40    8F  88 0003F          BISB2   #64, SETFILE$FLAGS+2                ; 2296
                                       7F  11 00044          BRB     7$                                  ;
          00000000G     8F             52  D1 00046 1$:      CMPL    STATUS, #LIB$_QUICONACT             ; 2297
                                       0A  12 0004D          BNEQ    2$                                  ;
               02       A3       80    8F  88 0004F          BISB2   #128, SETFILE$FLAGS+2               ; 2298
                                 52    01  D0 00054          MOVL    #1, STATUS                          ;
                                       1A  11 00057          BRB     3$                                  ;
          00000000G     8F             52  D1 00059 2$:      CMPL    STATUS, #LIB$_NEGANS                ; 2299
                                       11  13 00060          BEQL    3$                                  ;
                                       52  DD 00062          PUSHL   STATUS                              ; 2300
                                 018C  C3  9F 00064          PUSHAB  CONF_DESC                           ;
                                       01  DD 00068          PUSHL   #1                                  ;
                        00000000G      8F  DD 0006A          PUSHL   #SET$_WRITEERR                      ;
                     64                04  FB 00070          CALLS   #4, LIB$SIGNAL                      ;
                     4F                52  E9 00073 3$:      BLBC    STATUS, 7$                          ; 2302
                     51          16    A3  9E 00076 4$:      MOVAB   GLOBAL_MASK, P_RES_MASK             ; 2312
                     50          14    A3  9E 0007A          MOVAB   GLOBAL_PROT, P_RES_PROT             ; 2313
                                 16    A3  B5 0007E          TSTW    GLOBAL_MASK                         ; 2315
                                       02  12 00081          BNEQ    5$                                  ;
                                       50  7C 00083          CLRQ    P_RES_PROT                          ; 2316
                                 04    AE  9F 00085 5$:      PUSHAB  FINAL_PROT                          ; 2323
                                       50  DD 00088          PUSHL   P_RES_PROT                          ; 2326
                                       51  DD 0008A          PUSHL   P_RES_MASK                          ; 2325
                                 018C  C3  9F 0008C          PUSHAB  CONF_DESC                           ; 2323
          00000000G     00             04  FB 00090          CALLS   #4, LIB$SET_FILE_PROT               ;
                        52             50  D0 00097          MOVL    R0, STATUS                          ;
                        15             52  E8 0009A          BLBS    STATUS, 6$                          ;
                                       52  DD 0009D          PUSHL   STATUS                              ; 2335
                                 018C  C3  9F 0009F          PUSHAB  CONF_DESC                           ; 2331
                                       01  DD 000A3          PUSHL   #1                                  ;
                        00000000G      8F  DD 000A5          PUSHL   #SET$_PRONOTCHG                     ;
                     64                04  FB 000AB          CALLS   #4, LIB$SIGNAL                      ;
                     50                52  D0 000AE          MOVL    STATUS, R0                          ; 2336
                                       04  000B1            RET                                         ;
OE      01              A3             04  E1 000B2 6$:      BBC     #4, SETFILE$FLAGS+1, 7$             ; 2344
                        7E       04    AE  3C 000B7          MOVZWL  FINAL_PROT, -(SP)                   ; 2345
```

```
                                  04    AC  DD 000BB        PUSHL   FAB                         ;
                       00000000V  EF    02  FB 000BE        CALLS   #2, PROT_LOG_RESULTS        ;
                                  50    01  D0 000C5 7$:    MOVL    #1, R0                      ; 2348
                                            04 000C8        RET                                 ; 2350

; Routine Size:  201 bytes,    Routine Base:  $CODE$ + 142F
```

```
;  2365     2351  1 ROUTINE prot_log_results (fab,final_prot): =
;  2366     2352  1
;  2367     2353  1 !----
;  2368     2354  1 !
;  2369     2355  1 ! Functional description
;  2370     2356  1 !
;  2371     2357  1 !       This routine is called from SETPRO_ACTION whenever
;  2372     2358  1 !       logging for /PROTECTION is requested
;  2373     2359  1 !
;  2374     2360  1 ! Input parameters
;  2375     2361  1 !
;  2376     2362  1 !       fab = Address of block describing the file
;  2377     2363  1 !       fab$l_nam = pointer to name block
;  2378     2364  1 !
;  2379     2365  1 ! Output parameters
;  2380     2366  1 !
;  2381     2367  1 !       First error encountered, or TRUE is RETURNed
;  2382     2368  1 !
;  2383     2369  1 !----
;  2384     2370  1
;  2385     2371  2 BEGIN
;  2386     2372  2
;  2387     2373  2 LITERAL
;  2388     2374  2     pbufsize = 32;                        ! Buffer for generating string
;  2389     2375  2
;  2390     2376  2 MAP fab: REF $BBLOCK;                     ! Define fab block format
;  2391     2377  2
;  2392     2378  2 BIND nam = .fab[fab$l_nam]: $BBLOCK;      ! Define name block
;  2393     2379  2
;  2394     2380  2 LOCAL
;  2395     2381  2     status:,                              ! Recieves status
;  2396     2382  2     pbuf:       VECTOR[pbufsize,BYTE],    ! Place for protection string
;  2397     2383  2     pdesc:      VECTOR[2],                ! Temporary string descriptor
;  2398     2384  2     desc:       VECTOR[2],                ! Temporary string descriptor
;  2399     2385  2     prot_table: VECTOR[4];                ! Protection string table
;  2400     2386  2
;  2401     2387  2
;  2402     2388  2 !
;  2403     2389  2 ! Now build the resultant protection string from the value passed
;  2404     2390  2 ! in the call.
;  2405     2391  2 !
;  2406     2392  2
;  2407     2393  2 expand_prot(                              ! Call sub with
;  2408     2394  2         prot_table,                       !  -place for the result
;  2409     2395  2         .final_prot);                     !  -final protection value
;  2410     2396  2
;  2411     2397  2 pdesc[0] = pbufsize;                      ! Initialize descriptor size
;  2412     2398  2 pdesc[1] = pbuf;                         ! Initialize descriptor address
;  2413     2399  2
;  2414     2400  3 IF NOT (status =
;  2415   P 2401  3 $FAOL    (                               ! Call system service with
;  2416   P 2402  3         CTRSTR = ADDRDESC ('S:!AS,O:!AS,G:!AS,W:!AS'),  ! -FAO string
;  2417   P 2403  3         OUTLEN = pdesc[0],              !  -place for resultant length
;  2418   P 2404  3         OUTBUF = pdesc,                 !  -output buffer descriptor
;  2419     2405  4         PRMLST = prot_table)            !  -address of list of args
;  2420     2406  3 ) THEN BEGIN
;  2421     2407  3         SIGNAL (.status);               ! Oops, tell the user
```

```
;  2422          2408  3         return (.status);                ! And exit immediately
;  2423          2409  3         END;
;  2424          2410
;  2425          2411  2     SIGNAL          (set$_protected,      ! Inform user with
;  2426          2412                          2,                  !  -two FAO arguements
;  2427          2413                          conf_desc,          !  -file name
;  2428          2414                          pdesc);             !  -new protection
;  2429          2415
;  2430          2416  2 RETURN (true);
;  2431          2417  2
;  2432          2418  1 END;


                                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

21  3A  47  2C  53  41  21  3A  4F  2C  53  41  21  3A  53  004A8 P.AEG:  .ASCII  \S:!AS,O:!AS,G:!AS,W:!AS\<0>                ;
                    00  53  41  21  3A  57  2C  53  41  004B7                                                              ;
                                            00000017  004C0 P.AEF:  .LONG   23                                           ;
                                            00000000'  004C4        .ADDRESS P.AEG                                      ;

                                                                    .EXTRN  SYS$FAOL

                                                                    .PSECT  $CODE$,NOWRT,2

                                                 000C 00000 PROT_LOG_RESULTS:
                                                            .WORD   Save R2,R3                                          ; 2351
                                    53 00000000G   00  9E 00002     MOVAB   LIB$SIGNAL, R3                              ;
                                    5E       C0    AE  9E 00009     MOVAB   -64(SP), SP                                 ;
                                             08    AC  DD 0000D     PUSHL   FINAL_PROT                                  ; 2395
                                             04    AE  9F 00010     PUSHAB  PROT_TABLE                                  ; 2393
                                00000000V    EF        02  FB 00013     CALLS   #2, EXPAND_PROT
                                      18     AE        20  DO 0001A     MOVL    #32, PDESC                              ; 2397
                                      1C     AE    20  AE  9E 0001E     MOVAB   PBUF, PDESC+4                           ; 2398
                                                 5E  DD 00023     PUSHL   SP                                          ; 2405
                                                 1C  AE  9F 00025     PUSHAB  PDESC
                                                 20  AE  9F 00028     PUSHAB  PDESC
                                          00000000'  EF  9F 0002B     PUSHAB  P.AEF
                                00000000G    00        04  FB 00031     CALLS   #4, SYS$FAOL
                                             52        50  DO 00038     MOVL    R0, STATUS
                                             09        52  E8 0003B     BLBS    STATUS, 1$
                                                       52  DD 0003E     PUSHL   STATUS                                  ; 2407
                                             63        01  FB 00040     CALLS   #1, LIB$SIGNAL
                                             50        52  DO 00043     MOVL    STATUS, R0                              ; 2408
                                                       04  00046     RET
                                                 18  AE  9F 00047 1$:  PUSHAB  PDESC                                  ; 2411
                                          00000000'  EF  9F 0004A     PUSHAB  CONF_DESC
                                                 02  DD 00050     PUSHL   #2
                                          00000000G  8F  DD 00052     PUSHL   #SET$_PROTECTED
                                             63        04  FB 00058     CALLS   #4, LIB$SIGNAL
                                             50        01  DO 0005B     MOVL    #1, R0                                  ; 2416
                                                       04  0005E     RET                                             ; 2418

; Routine Size:  95 bytes,    Routine Base:  $CODE$ + 14F8
```

```
;  2434     2419  1 ROUTINE expand_prot (table, protection): =
;  2435     2420  1
;  2436     2421  1 !----
;  2437     2422  1 !
;  2438     2423  1 ! Functional description
;  2439     2424  1 !
;  2440     2425  1 !       This routine, called from PROT_LOG_RESULTS, fills
;  2441     2426  1 !       a given VECTOR with the addresses of strings
;  2442     2427  1 !       corresponding to a given protection word.
;  2443     2428  1 !
;  2444     2429  1 ! Input parameters
;  2445     2430  1 !
;  2446     2431  1 !       table = Address of the table to be filled in.
;  2447     2432  1 !       protection = Protection word.
;  2448     2433  1 !
;  2449     2434  1 ! Output parameters
;  2450     2435  1 !
;  2451     2436  1 !       table has been filled in with the addresses of descriptors
;  2452     2437  1 !       of strings describing each type of user (SYS,OWN,GRP,WORLD).
;  2453     2438  1 !
;  2454     2439  1 !----
;  2455     2440  1
;  2456     2441  2 BEGIN
;  2457     2442  2
;  2458     2443  2 BIND
;  2459     2444  2     prot_table = .table: VECTOR[4];      ! Table of addresses
;  2460     2445  2
;  2461     2446  2 OWN
;  2462     2447  2     prot_values: VECTOR[16] INITIAL(     ! Protection descriptions
;  2463     2448  2             ADDRDESC('RWED'),
;  2464     2449  2             ADDRDESC('WED'),
;  2465     2450  2             ADDRDESC('RED'),
;  2466     2451  2             ADDRDESC('ED'),
;  2467     2452  2             ADDRDESC('RWD'),
;  2468     2453  2             ADDRDESC('WD'),
;  2469     2454  2             ADDRDESC('RD'),
;  2470     2455  2             ADDRDESC('D'),
;  2471     2456  2             ADDRDESC('RWE'),
;  2472     2457  2             ADDRDESC('WE'),
;  2473     2458  2             ADDRDESC('RE'),
;  2474     2459  2             ADDRDESC('E'),
;  2475     2460  2             ADDRDESC('RW'),
;  2476     2461  2             ADDRDESC('W'),
;  2477     2462  2             ADDRDESC('R'),
;  2478     2463  2             ADDRDESC(''));
;  2479     2464  2
;  2480     2465  2 INCR index FROM 0 TO 3 DO
;  2481     2466  2     prot_table[.index] = .prot_values [.protection<.index*4,4>];
;  2482     2467  2
;  2483     2468  2 RETURN (true);           ! Always return true
;  2484     2469  2
;  2485     2470  1 END;

                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2
```

```
                    44  45  57  52  004C8 P.AEI:   .ASCII  \RWED\
                      00000004  004CC P.AEH:   .LONG   4
                      00000000' 004D0          .ADDRESS P.AEI
                    00  44  45  57  004D4 P.AEK:   .ASCII  \WED\<0>
                      00000003  004D8 P.AEJ:   .LONG   3
                      00000000' 004DC          .ADDRESS P.AEK
                    00  44  45  52  004E0 P.AEM:   .ASCII  \RED\<0>
                      00000003  004E4 P.AEL:   .LONG   3
                      00000000' 004E8          .ADDRESS P.AEM
                    00  00  44  45  004EC P.AEO:   .ASCII  \ED\<0><0>
                      00000002  004F0 P.AEN:   .LONG   2
                      00000000' 004F4          .ADDRESS P.AEO
                    00  44  57  52  004F8 P.AEQ:   .ASCII  \RWD\<0>
                      00000003  004FC P.AEP:   .LONG   3
                      00000000' 00500          .ADDRESS P.AEQ
                    00  00  44  57  00504 P.AES:   .ASCII  \WD\<0><0>
                      00000002  00508 P.AER:   .LONG   2
                      00000000' 0050C          .ADDRESS P.AES
                    00  00  44  52  00510 P.AEU:   .ASCII  \RD\<0><0>
                      00000002  00514 P.AET:   .LONG   2
                      00000000' 00518          .ADDRESS P.AEU
                    00  00  00  44  0051C P.AEW:   .ASCII  \D\<0><0><0>
                      00000001  00520 P.AEV:   .LONG   1
                      00000000' 00524          .ADDRESS P.AEW
                    00  45  57  52  00528 P.AEY:   .ASCII  \RWE\<0>
                      00000003  0052C P.AEX:   .LONG   3
                      00000000' 00530          .ADDRESS P.AEY
                    00  00  45  57  00534 P.AFA:   .ASCII  \WE\<0><0>
                      00000002  00538 P.AEZ:   .LONG   2
                      00000000' 0053C          .ADDRESS P.AFA
                    00  00  45  52  00540 P.AFC:   .ASCII  \RE\<0><0>
                      00000002  00544 P.AFB:   .LONG   2
                      00000000' 00548          .ADDRESS P.AFC
                    00  00  00  45  0054C P.AFE:   .ASCII  \E\<0><0><0>
                      00000001  00550 P.AFD:   .LONG   1
                      00000000' 00554          .ADDRESS P.AFE
                    00  00  57  52  00558 P.AFG:   .ASCII  \RW\<0><0>
                      00000002  0055C P.AFF:   .LONG   2
                      00000000' 00560          .ADDRESS P.AFG
                    00  00  00  57  00564 P.AFI:   .ASCII  \W\<0><0><0>
                      00000001  00568 P.AFH:   .LONG   1
                      00000000' 0056C          .ADDRESS P.AFI
                    00  00  00  52  00570 P.AFK:   .ASCII  \R\<0><0><0>
                      00000001  00574 P.AFJ:   .LONG   1
                      00000000' 00578          .ADDRESS P.AFK
                                0057C P.AFM:   .BLKB   0
                      00000000  0057C P.AFL:   .LONG   0
                      00000000' 00580          .ADDRESS P.AFM

                                               .PSECT  $OWN$,NOEXE,2

00000000' 00000000' 00000000' 00000000' 00000000' 00000000' 00034 PROT_VALUES:
                                               .ADDRESS P.AEH, P.AEJ, P.AEL, P.AEN, P.AEP, -
00000000' 00000000' 00000000' 00000000' 00000000' 00000000' 0004C                P.AER, P.AET, P.AEV, P.AEX, P.AEZ, P.AFB, -
                    00000000' 00000000' 00000000' 00000000' 00064                P.AFD, P.AFF, P.AFH, P.AFJ, P.AFL
```

```
                                                              .PSECT  $CODE$,NOWRT,2

                                          0004 00000 EXPAND_PROT:
                                                              .WORD   Save R2                                    ; 2419
                                        50  D4 00002          CLRL    INDEX                                      ; 2465
                 52          50         02  78 00004 1$:      ASHL    #2, INDEX, R2                              ; 2466
     51      08  AC          04         52  EF 00008          EXTZV   R2, #4, PROTECTION, R1                     ;
                       04 BC40 00000000'EF41  D0 0000E        MOVL    PROT_VALUES[R1], @TABLE[INDEX]             ;
                 E8          50         03  F3 00018          AOBLEQ  #3, INDEX, 1$                              ;
                             50         01  D0 0001C          MOVL    #1, R0                                     ; 2468
                                            04 0001F          RET                                                ; 2470

; Routine Size:  32 bytes,     Routine Base:  $CODE$ + 1557
```

```
;  2487          2471  1 ROUTINE parse_class (desc) =
;  2488          2472  1
;  2489          2473  1 !---
;  2490          2474  1 !
;  2491          2475  1 ! This routine called from SETPRO_ACTION, parses one class of user
;  2492          2476  1 ! (e.g. SYSTEM, OWNER, GROUP, WORLD) to see what protection is allowed.
;  2493          2477  1 ! The value returned in the low 4 bits is the protection code, with the
;  2494          2478  1 ! bits set to reflect that access is  requested.  Note that this is
;  2495          2479  1 ! exactly the opposite of what the system wants.
;  2496          2480  1 !
;  2497          2481  1 ! Inputs:
;  2498          2482  1 !
;  2499          2483  1 !       DESC -- a descriptor pointing to the ASCII representation of the
;  2500          2484  1 !               protection desired
;  2501          2485  1 !
;  2502          2486  1 !---
;  2503          2487  1
;  2504          2488  2 BEGIN
;  2505          2489  2
;  2506          2490  2 MAP desc : REF $BBLOCK;
;  2507          2491  2
;  2508          2492  2 LOCAL
;  2509          2493  2     pointer,                          ! Pointer to string
;  2510          2494  2     result;                           ! Resultant protection
;  2511          2495  2
;  2512          2496  2 !
;  2513          2497  2 ! Initially set the value to all zeros, no access
;  2514          2498  2 !
;  2515          2499  2 result = 0;
;  2516          2500  2
;  2517          2501  2 !
;  2518          2502  2 ! Scan for the occurrence of each keyletter, and, if it is there, set the
;  2519          2503  2 ! appropriate bit.
;  2520          2504  2 !
;  2521          2505  2 pointer = .desc[dsc$a_pointer];
;  2522          2506  2 INCR index FROM 1 to .desc[dsc$w_length] DO
;  2523          2507  2     BEGIN
;  2524          2508  2     LOCAL char : BYTE;
;  2525          2509  2     char = CH$RCHAR_A(pointer);
;  2526          2510  2     IF .char EQL'R'
;  2527          2511  2     THEN result = .result OR %X'1'
;  2528          2512  2     ELSE IF .char EQL  'W'
;  2529          2513  2     THEN result = .result OR %X'2'
;  2530          2514  2     ELSE IF .char EQL 'E'
;  2531          2515  2          OR .char EQL 'P'
;  2532          2516  2     THEN result = .result OR %X'4'
;  2533          2517  2     ELSE IF .char EQL 'D'
;  2534          2518  2          OR .char EQL 'L'
;  2535          2519  2     THEN result = .result OR %X'8'
;  2536          2520  2     ELSE SIGNAL_STOP (set$_syntax, 1, .desc);
;  2537          2521  2     END;
;  2538          2522  2
;  2539          2523  2 RETURN .result;
;  2540          2524  1 END;
```

```
                                007C 00000 PARSE_CLASS:
                                           .WORD   Save R2,R3,R4,R5,R6                     ; 2471
            52      04   AC    D0 00002           MOVL    DESC, R2                        ; 2505
            56      04   A2    D0 00006           MOVL    4(R2), POINTER
            55           62    3C 0000A           MOVZWL  (R2), R5                        ; 2506
                         53    7C 0000D           CLRQ    INDEX
                         4C    11 0000F           BRB     8$
            50           86    90 00011 1$:       MOVB    (POINTER)+, CHAR                ; 2509
      52    8F           50    91 00014           CMPB    CHAR, #82                       ; 2510
                         05    12 00018           BNEQ    2$
            54           01    88 0001A           BISB2   #1, RESULT                      ; 2511
                         3E    11 0001D           BRB     8$
      57    8F           50    91 0001F 2$:       CMPB    CHAR, #87                       ; 2512
                         05    12 00023           BNEQ    3$
            54           02    88 00025           BISB2   #2, RESULT                      ; 2513
                         33    11 00028           BRB     8$
      45    8F           50    91 0002A 3$:       CMPB    CHAR, #69                       ; 2514
                         06    13 0002E           BEQL    4$
      50    8F           50    91 00030           CMPB    CHAR, #80                       ; 2515
                         05    12 00034           BNEQ    5$
            54           04    88 00036 4$:       BISB2   #4, RESULT                      ; 2516
                         22    11 00039           BRB     8$
      44    8F           50    91 0003B 5$:       CMPB    CHAR, #68                       ; 2517
                         06    13 0003F           BEQL    6$
      4C    8F           50    91 00041           CMPB    CHAR, #76                       ; 2518
                         05    12 00045           BNEQ    7$
            54           08    88 00047 6$:       BISB2   #8, RESULT                      ; 2519
                         11    11 0004A           BRB     8$
                         52    DD 0004C 7$:       PUSHL   R2                              ; 2520
                         01    DD 0004E           PUSHL   #1
                  007710FA   8F    DD 00050           PUSHL   #7803130
      00000000G   00               03    FB 00056           CALLS   #3, LIB$STOP
B0                53               55    F3 0005D 8$:       AOBLEQ  R5, INDEX, 1$           ; 2506
                  50               54    D0 00061           MOVL    RESULT, R0              ; 2523
                                         04 00064           RET                             ; 2524

; Routine Size:  101 bytes,    Routine Base:  $CODE$ + 1577
```

```
; 2542          2525  1 END
; 2543          2526  0 ELUDOM


                                                        .EXTRN  LIB$SIGNAL, LIB$STOP

;                               PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $GLOBAL$                       1112  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                           116  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                         1412  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                         5596  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  .  ABS  .                         0  NOVEC,NOWRT,NORD ,NOEXE,NOSHR,  LCL,  ABS,  CON,NOPIC,ALIGN(0)


;                           Library Statistics
;
;                                    -------- Symbols --------      Pages       Processing
;       File                         Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1      18619       224         1        1000        00:01.9
;  _$255$DUA28:[SYSLIB]CLIMAC.L32;1      14         0         0           9        00:00.1




;                                       COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:SETFILE/OBJ=OBJ$:SETFILE MSRC$:SETFILE/UPDATE=(ENH$:SETFILE)

; Size:          5596 code + 2640 data bytes
; Run Time:         01:41.5
; Elapsed Time:     05:29.3
; Lines/CPU Min:     1492
; Lexemes/CPU-Min: 22862
; Memory Used:  780 pages
; Compilation Complete
```

SETFILE
LIS

SETP0MESS
LIS

SETP0DISP
LIS

SETMISC
LIS

SETPRO
LIS

SETMAIN
LIS